DEPARTMENT OF OCEAN ENGINEERING
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS 02139

APPLICATION OF COMPUTER AIDED DESIGN TO SUBMARINES

by

MARVIN ERNEST MEADE II

O.E.
S.M. (NA&ME)

COURSE 13A
MAY 1984

Thesis
M39954

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93943

APPLICATION OF COMPUTER AIDED DESIGN TO SUBMARINES

by

MARVIN ERNEST MEADE II

B.S., Michigan Technological University
(1976)

Submitted to the Department of
Ocean Engineering
in Partial Fulfillment of the
Requirements of the Degrees of

OCEAN ENGINEER

and

MASTER OF SCIENCE IN
NAVAL ARCHITECTURE AND MARINE ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1984

© MARVIN ERNEST MEADE II 1984

The author hereby grants M.I.T. and agencies of the U.S. Navy permission to reproduce and to distribute copies of this thesis in whole or in part.

Thesis
M39954
c.1

APPLICATION OF COMPUTER AIDED DESIGN

TO SUBMARINES

by

MARVIN ERNEST MEADE II

Submitted to the Department of Ocean Engineering on May 11, 1984 in Partial Fulfillment of the Requirements of the Degrees of Ocean Engineer and Master of Science in Naval Architecture and Marine Engineering

ABSTRACT

A software package was developed for use as a design tool in the conceptual design of submarines on the Computervision CGP-200X Designer System, a turnkey computer aided design hardware and graphics software system. The philosophy behind the software package is to keep all major design decisions under control of the design engineer rather than embedding them within program algorithms.

Modules are provided for generating three dimensional models of the outer envelope and sail, and for the interactive design of the pressure hull. This package is designed to interface with modules developed separately by Patrick Hale which calculate weight estimates, principal characteristics, envelope geometry, resistance, balance, and the equilibrium polygon. Interactive graphics are used extensively.

The software requires a knowledgeable naval architect as the user, but does not require an extensive knowledge of computers or computer aided design systems.

Thesis Supervisor: Dr. David V. Burke

Title: Professor of Ocean Engineering

DUDLEY KNOX LIBRARY

TABLE OF CONTENTS

ACKNOWLEDGMENTS

The writing of this thesis, which concerns computer aided design, would have been much more difficult and much less interesting if it were not possible to experiment with an appropriate system. Mr. Richard Salter of Computervision Corporation arranged unlimited access to a CGP-200X at Computervision's Bedford, Massachusetts facility. Having access to a system, it also helps if one knows how it operates. Manuals help, but they usually do not provide answers to every problem which arises. In all probability this thesis would not have progressed as much as it did without the assistance of Mr. James Kelly, also of Computervision, who answered countless questions about the system and its programming.

The author would also like to thank Professor David Burke for serving as thesis supervisor.

I. INTRODUCTION

Computer aided design has become an important tool in many industries over the last ten years. The capabilities of CAD systems, like those of other computer systems, are growing at a phenomenal rate. CAD is now becoming the standard mode of design for major architecture firms and the automotive and aerospace industries.

Computer aided design has been accepted more slowly in the area of naval ship design. The comparatively low production rates of naval vessels makes the cost effectiveness of CAD a greater question for ship design agencies. The scale and complexity of naval combatants makes the design of a flexible and comprehensive software package a formidable task.

A number of software packages have been implemented for conceptual and preliminary design of surface combatants in both government and private sector design organizations. The majority of these tend to be batch oriented ship synthesis programs with little or no real time interactive design capability.

This lack of interactive graphics capability is due to a number of factors. The concern for standardization and the lengthy process of government certification and

procurement create a substantial delay between the initial availability and subsequent installation of new hardware. Computer aided design, despite its acceptance in some major industries, is nevertheless unproven in the world of naval architecture. Many design procedures used in naval architecture are somewhat subjective and difficult to quantify.

A further impediment to the use of CAD in naval architecture has been the tendency of many early software packages to embed many key decisions in program algorithms, thus taking the designer out of the decision loop. Such software does not allow the designer to utilize his knowledge and expertise and frequently precludes trading off design parameters to achieve optimal designs.

In the area of submarine design the above problems are more acute. The number of new submarine designs is even less than the number of surface combatants. Accuracy is of more importance in submarine design. Where feet and tons may be acceptable limits of tolerance in a large surface combatant inches and pounds frequently determine the limits of accuracy in submarine design. This is due to the requirement to achieve a hydrostatic balance in both the submerged and surfaced conditions, as well as the premium attached to internal volume and

deck area.

Submarine designers are fewer in number than their surface counterparts, and often have many design relationships tied to subjective criteria developed over years of experience. Software which does not allow the designer to make use of this subjective design flexibility is not likely to be accepted by the design community.

The software developed in this thesis, titled CADSUB (Computer Aided Design of Submarines), accomplishes a first pass through the submarine design process at the conceptual level of design. In conjunction with the Ocean Engineer thesis written by Pat Hale, the complete package includes modules for the calculation of weight estimates, envelope and pressure hull geometry, speed and power, basic hydrostatics, weight and moment balance, and the determination of the equilibrium polygon.

A primary design philosophy for the software was to develop it for use by a competent naval architect who is not necessarily an expert in the use of computers. Critical design decisions are left to the user rather than being embedded in program code. Opportunities are provided to override program algorithms where necessary to implement the desires of the individual designer. All

communications between the user and the program utilize standard naval architecture terminology, making it unnecessary for the user to learn specific computer terminology. In addition, if the designer is familiar with the operation of the CAD system he may easily exit the program at appropriate points and tailor the design to provide more detail or alternate geometry and analytical procedures. Interactive graphics are used to provide a clear visualization of the design as it develops.

II. USES OF COMPUTER AIDED DESIGN

The communication of ideas and information is an essential element of all engineering endeavors. Historically the primary medium used for communication and storage of information has been paper. Paper has the advantage of allowing both long term storage and the distribution of information to a large number of users at a moderate cost. However, information, once committed to paper, is sometimes difficult to modify.

The advent of modern computer aided design technology offers the opportunity to improve the design process by removing some of the limitations imposed by paper. Using a CAD system, a designer may develop, analyze, and refine ideas and solutions to engineering problems in a fraction of the time required using paper and hand calculations. Once a design exists in the form of a three dimensional model in a computer data base, it may be recalled, displayed, and modified in a manner analogous to the way in which documents may be edited by a word processor. During the evolution of a design over a period of several years with input from numerous designers, the use of a CAD system can facilitate storage of all information pertaining to a design in a

central location, ensuring adequate access to this information for everyone involved in the design process.

The term computer aided design has been in existence for approximately thirty years. The earliest applications of computer aided design were limited to two major areas - electronic drafting with computer controlled plotting pens and computer programs used to process numerical data. Modern CAD systems integrate a wide variety of functions including geometric modeling, simulation, engineering analysis, testing, interfacing directly with the manufacturing process, and automated drafting. These capabilities allow manufacturers to build solid models of engines, road test automobiles, and examine the stability of offshore structures on computer screens prior to construction of prototypes or models.

The foundation of computer aided design techniques is the computer representation of a part's size and shape, known as the geometric model. The simplest representations of models are wire frame drawings which depict the outline of the shape with a simple mesh. With improvements in CAD technology it is increasingly more common to use the more realistic looking solid modeling technique to provide visual representation because it more thoroughly specifies each facet of the object being

profiled.

Solid models are handled by CAD systems almost as if they were real objects. Models may be inverted, spun, or cut. They may be used to calculate physical properties such as weight and center of gravity. Now, in a matter of a few minutes or hours, it is possible for an engineer to ascertain the week spots in a structural design or the behavior of a dynamic system.

Computer aided design has become firmly entrenched as a valuable tool in both the automobile and aerospace industries. As a result, the design process in these industries has been speeded up significantly. Other industries, such as shipbuilding, are now beginning to accept computer aided design.

GEOMETRIC MODELING

The process of creating a computer based representation of a design is the most basic application of modern computer aided design. Using the graphics capabilities inherent in a CAD system, the design engineer creates a three dimensional model of a new part. This model may then be rotated or cut in cross-section to expose exterior and interior details. Other parts which exist in the data base may be called up and added to the part being designed. In this manner the

interaction and fit between parts and systems may be checked and designs modified to allow for interference before the manufacturing process begins.

ANALYSIS OF DESIGNS

In-service behavior of a model may be predicted with finite element analysis. Several developments in the last few years have made the use of finite element analysis more useful and less time consuming. Automated mesh generation is among the most important of these developments.

At one time data for finite element programs were written out by hand and then keypunched into a computer for batch processing. This resulted in large quantities of numerical printouts which required an experienced analyst to evaluate. With the advent of preprocessing techniques the operator could interactively construct the finite element mesh directly on the CRT screen.

The latest trend in finite element analysis is automatic mesh generation. Using this technique the operator divides the part shape into coarse four sided areas. Mesh density is then specified and nodes are automatically positioned by the computer. Some modeling programs exist which use free mesh generating techniques, making it possible to create meshes for

parts with irregular curves, internal cutouts, and intricate details. Automatic mesh generation makes it possible to conduct analysis of parts using a finer mesh than possible with manually generated meshes. Furthermore, costs are considerably less when automated mesh generation is used since labor costs are significantly reduced.

Following creation of the mesh and analysis by a finite elements program postprocessing routines are used to interpret the results and reduce the need to analyze voluminous numerical data. Contour plots may be used to indicate lines of constant strain, or color may be used to differentiate between different levels of stress. Deformed shape plots may be generated to show model displacement.

Finite element techniques are routinely applied to stress analysis problems. They may also be applied to a wide range of other problems including fluid mechanics, heat flow, acoustics, and electromagnetics.

SIMULATION

An application of computer aided design which is becoming increasingly more popular is that of simulation. When used in conjunction with geometric modeling and finite element analysis, simulation may be used to

preview the dynamic behavior of a design in service.

Simulation programs now being introduced permit the designer to profile in three dimensions the behavior of complex equipment or machinery. A single part may be studied as a component of a dynamic system in order to assess the loads it is likely to encounter. Finite element analysis may then be employed to determine stress distribution.

Design software packages generally include capability for simulating simple forms of mechanical behavior. These kinematic programs mimic the paths of cams, gears, links, and other mechanisms which undergo cyclic or repetitve movement. A number of turnkey systems are capable of animating the motion of simple pivoted devices such as hinges, doors, or cranks. Programs for simulating more complex mechanisms are usually obtained as unbundled software.

Systems undergoing large displacements may be simulated using displacement analysis programs. Those systems which have been analyzed by this technique include the suspension system of an automobile as it makes a turn, nuclear reactor handling mechanisms, the movement of control linkages and robots, high-speed mechanisms, and construction equipment. [1]

BENEFITS OF COMPUTER AIDED DESIGN

The acquisition, maintenance, and training costs of a CAD system constitutes a significant capital investment. It must, therefore, provide substantial benefits to justify its cost. This is not a simple evaluation to make since there is not likely to be a reduction in design costs following installation of a CAD system. Savings are more often realized in indirect ways which are difficult to quantify.

The principal benefit of computer aided design is likely to be in the areas of cost avoidance and risk reduction. Computer aided design, because it speeds the design process, allows more time to explore design alternatives. Several companies which have made substantial use of CAD systems have proven that the time required to complete individual design tasks may be reduced significantly. For example, prior to the implementation of CAD, engineers at Pontiac took up to six weeks to design a cylinder head combustion chamber by manually calculating complex geometric constructions. Using CAD, this task may now be completed in three days. Since the time to complete an individual design is reduced, the time available for design optimization and study of alternatives is increased.

The ability to model several parts in a design and study their interaction with one another and with other systems makes it much easier to design around interferences. This reduces the amount of rework required during production.

By reducing the time required for individual design tasks, providing time to study alternatives, and reducing rework, studies show that cost overruns and redesign costs may be reduced by as much as 50%. [2]

CHAPTER SUMMARY

This chapter has briefly outlined the major capabilities and applications of computer aided design. The implementation of CAD has the potential, if properly managed, to increase both the productivity and effectiveness of the design process, and thus improve the productivity and quality of the manufacturing process.

Computer aided design must not, however, be viewed as a replacement for a qualified designer. The CAD system is operated by a human, and the designs generated on the CAD system, as with any other computer system, are only as good as the data on which they are based.

III. DESCRIPTION OF THE COMPUTERVISION CGP-200X DESIGNER SYSTEM

When developing computer aided design software, particularly software which relies heavily on graphics, access to a CAD system is extremely helpful. The system used during the course of this thesis was the Computervision CGP-200X Designer System. This chapter contains a brief description of the Designer System.

The CGP-200X Designer System is a dedicated CAD system consisting of the following major components:

- CGP-200X Graphics Processor
- Storage Module
- Instaview Workstation
- CAD Software
- System Plotters

The CGP-200X Graphics Processor processes data and performs calculations related to the overall operation of the CAD graphics system. It consists of a graphics processor and a magnetic tape unit. The graphics processor is a high-speed data manipulation and storage device which controls all Designer System operations. Associated with the graphics processor's central processor unit are two memory boards, each capable of

storing either 128K or 512K, 16-bit words. The magnetic tape unit stores design information entered into the system on magnetic tape so that it may be transferred to other systems or maintained as a backup copy. [3]

The Storage Module, or disk unit, stores design information onto a multi-surfaced 300 megabyte disk pack. The operating system (OS) resides in a protected portion of the primary disk. Multiple disk packs may be supported, providing additional disk storage space and the possibility of increased disk and file access speed.

Communication between the design engineer and the Graphics Processor is achieved with the Instaview Workstation. Components making up the Instaview Workstation include the following:

- Graphics display screen
- Graphics tablet, electronic pen, and menu
- Alphanumeric keyboard
- Image Control Unit
- Workstation printer

The graphics display screen allows the user to view model geometry and detail drawings as they are being constructed. In addition, the screen also displays the text of commands used to construct graphic elements. This text occupies an area of either four or twenty-four lines of eighty columns.

Within the graphics area of the display screen entry of graphic information is managed by a cursor. It is used to locate positions on the screen to be digitized and to identify existing graphics. For example, if a line is to be drawn between two points, the cursor is used to locate first one point to be digitized and then the other. The cursor may be controlled either with the electronic pen and graphics tablet or by direct entry of data points with the alphanumeric keyboard.

The graphics display screen may be either a monochrome or color raster scan CRT. The monochrome display is a nineteen inch, 1000 line, high-resolution, green phosphor CRT. Color graphics are provided on the Instaview C terminal. In addition to the same raster scan technology used by the monochrome display, up to sixty-four color combinations (seven basic colors and twenty-four shades which are definable by layers and views) may be used to reduce design time and increase accuracy by providing heightened graphics discrimination. [4]

Entry of of graphics information may be accomplished in two ways. In the event that exact locations are required, the keyboard may be used. The alternative is the graphics tablet and electronic pen. [5]

The graphics tablet has two distinct areas on its surface. These are the graphics area, which directly

corresponds to the graphics area on the display screen, and the menu area, which provides a fast means of entering graphics commands. The menu area consists of a number of small squares, sometimes referred to as function keys. Each square on the menu can represent a specific command, eliminating the need to type the command from the keyboard. A maximum of 512 squares may be programed to perform particular functions or a series of operations. [6]

The electronic pen is the activating device that locates positions or identifies existing graphic entities on the screen. Positions located, or digitized, generally define construction or coordinate information.

Like most computer systems, the Designer System is equipped with a keyboard for entry of alphanumeric information. It contains sixty-seven key positions, including upper and lower case, special symbol keys, and a numeric pad. The menu and keyboard may be used interchangeably.

Incorporated into the Instaview terminal is the Image Control Unit, or ICU. The ICU has twelve switches and eight buttons which perform various functions that dramatically affect the image displayed on the graphics screen. Functions controlled by the ICU include the display characteristics of text information (number of

lines) and graphics (brightness and darkness quality of background and graphics). In addition, the ICU allows dynamic manipulation of model geometry, making the construction and viewing of geometry easier. Dynamics capabilities provided by the ICU include enlarging or decreasing the apparent size of the model (zoom), moving the model in any direction of the graphics screen (scroll), and rotation of the model on any of the several available axes.

While generating a model or drawing on the Designer System, it is often desirable to make temporary working copies of the work in progress in a hardcopy format. This capability is provided by a desktop thermal dot matrix printer which serves as both a line printer and plotter, producing copies of both text and graphics. Once a model design or drawing has been completed high quality pen or dot matrix plots may be generated using one of several plotting devices.

Operation of the Computervision Designer System is controlled by CADDS 4X software. Within CADDS 4X two operational levels exist. These are the Operating System (OS) level and the Computer-Aided-Design and Drafting System (CADDS).

Operating System level activity is non-graphic in nature, and generally concerns the internal management

of CADDS 4X. Tasks performed at the OS level include creation and manipulation of data files, text files, and programs, as well as file transfer operations between the system disk and magnetic tape.

The second operational level, CADDS, is the level at which all graphics operations take place. Within this level there are two modes of operation; model mode and draw mode. These two modes exist because of the differences in thought processes normally associated with design and drafting. Model mode is used to create a representation of a real world three-dimensional object. When in model mode, the designer is creating an actual three dimensional layout of model geometry.

When constructing a model, one begins by activating a part. Next, a drawing is activated and one or more views are defined to display the model geometry. Using different views the designer may observe the model from a number of different viewpoints. Any number of drawings may be used to portray various aspects of the part. For example, one of the products of CADSUB is a drawing depicting a submarine outer envelope, pressure hull, and sail. By blanking from view various entities it is possible to obtain drawings which show only the outer envelope or only the pressure hull.

Drawing mode is used to generate drawing represen-

tations of the model by modifying or editing the model's geometry to change its visual appearance. Detail drawings with dimensions, text, and notes may be developed. When the drawing mode is used to edit the visual appearance of the model, changes that are made only affect the particular view in which the graphic modification takes place. This is in contrast with the model mode where changes made in one view are reflected in all views.

The CADDS 4X graphics was designed to be easy to learn and to use. It is an English based language, so it is not necessary for the operator to have significant knowledge of computers. Commands are entered by either typing them in on the keyboard or by using the menu. Traditional English spelling conventions are used for all commands, eliminating the need to learn computer commands and making the system easier to learn.

An extremely useful feature of the Designer System is the on-line documentation facility. This documentation is available to the operator at any time while in the graphics level, and allows the operator to obtain information about commands without leaving the workstation to consult a manual. A hard copy of this documentation may be made using either the workstation printer or the system line printer.

A series of commands or functions may be executed by using execute files. In those instances where a combination of computer graphics and non-graphic processing are required high level programming languages may be used to meet unique needs.

Several applications packages are available for the Designer System. These include Finite Element Modeling, Plant Design, Mechanical Design and Drafting, Wiring Diagrams, Printed Circuit/Electrical Schematic, Numerical Control, and others. Each of these packages enhance the utility of the Designer System in specific areas.

The primary purpose of this thesis is to develop a software package which utilizes the interactive graphics capabilities of the Computervision system, yet presents options and requests information in terms easily understood by a naval architect. This provides a useful design tool which requires only a minimum level of knowledge about the system to operate.

The Computervision Designer System is an extremely powerful tool for use in computer aided design and engineering. It is fairly easy to learn and become proficient with basic operation. However, like all computer systems, there is no substitute for experience if one expects to fully realize the Designer System's full potential.

This description of the Designer System is by no means complete. For a more thorough explanation of the system the reader is referred to the various Computervision manuals which provide detailed information about the system.

IV. CADSUB: THE APPLICATION OF COMPUTER AIDED DESIGN TO SUBMARINES

Before a new technology is universally accepted it must first be tested and proven to be useful and appropriate for the purpose for which it is intended. Computer aided design is no exception. The widespread acceptance of CAD by the automobile and aerospace industries may be viewed as an endorsement of its usefulness. It does not necessarily follow, however, that CAD is equally useful in all industries. In order to make a reasonably intelligent evaluation of the usefulness of CAD to the early stages of submarine design, the major focus of this thesis was the development of user interactive software which integrates both graphics and numerical calculations. Modules developed in this thesis create and display graphic displays of envelope geometry, aid in the design of the pressure hull, and demonstrate the techniques which may be used to model appendages. Appendix I contains a copy of the graphic output produced by these modules. Together with modules developed by Hale in his thesis "Computer Aided Conceptual Design of Submarines", these programs constitute a package of programs referred to as CADSUB. CADSUB is intended to

serve as a tool to aid in the conceptual design of submarines.

CADSUB is written in NEWVAR, a macro language available on the Computervision CGP-200X Designer System. NEWVAR allows the coupling of graphics commands with mathematical calculations in an execute file. This makes it possible to create a program which processes data and executes graphics commands for the user without requiring him to have an extensive knowledge of the graphics system. From the programming point of view, program development is simplified because graphics commands are similar to those used when manually manipulating graphics. It is not necessary to have the extensive knowledge of the internal workings of the computer.

Unfortunately, this convenience is not obtained without a price. NEWVAR's major disadvantage is its slowness, particularly when displaying text on the graphics screen. The handling of data files is complicated by the fact that all values written to or read from data files must be in the form of text variables. NEWVAR has no format statements, making it difficult to write organized output files. VARPRO2 is a newer Computervision macro language having some improved data transfer and formating features. It is not, however, subject to compilation before execution, and is thus significantly

slower than NEWVAR.

The modules developed for this thesis are contained in four NEWVAR programs. They are:

MIT.FORMC

MIT.ENV

MIT.PHULL

MIT.SAIL1

With the exception of MIT.FORMC each of these modules is designed to use data from either from direct interaction with the user or from pass files created in previously executed modules. Listings for each module are found in Appendix II. Simplified flowcharts of each module are in Appendix III. A description of data files created by CADSUB is included in Appendix IV, and layer descriptions are in Appendix V.

A. THE DRAWING FORM

The module MIT.FORMC is the only module not normally executed by CADSUB. It is used for the one time generation of a standard format drawing, or Form Part. This is a standard feature of the Computervision Designer System which allows creation of a standard drawing which may be used in any number of different designs.

A standard C size (17"x22") drawing format was

selected for the construction of this Form Part. The actual size of a hard copy plot may be scaled at the time the plot is made.

In addition to doing such mundane tasks as drawing a border and title block, three areas for the display of information are defined on the formatted drawing. The largest area is a table providing detailed pressure hull segment information in a spreadsheet format. Data displayed includes an identifying number for the segment, the forward and aftermost points of the segment, diameters at these locations, volume, and the location of the segments center of buoyancy. A maximum number of twenty pressure hull segments is allowed, and data for all twenty are displayed simultaneously.

The Pressure Hull Design Summary displays information concerning the pressure hull, which is created in the pressure hull module. Volume Required is the total volume of the pressure hull. This does not include volume reserved for air flasks, appendages, or other everbuoyant items. Volume Used is the volume occupied by pressure hull segments which make up the pressure hull at any given time. The Volume Remaining is simply the difference between the required volume and the volume used. The center of buoyancy of the entire pressure hull is also displayed.

The principle characteristics of the submarine are displayed in the third data display. Characteristics shown include length overall, diameter, the extents of the parallel midbody, submerged displacement and center of buoyancy, and the envelope displacement.

B. ENVELOPE GEOMETRY

A graphic display of the envelope geometry is created in the module MIT.ENV. Execution of this module is largely automatic, and requires the user only to specify the type of terminal (color or monochromatic) being used. The only reason for this interaction is the strange and inconsistent effects observed when using color commands on a monochrome terminal.

Following specification of terminal type the program obtains offsets and principal dimensions from pass files created in earlier modules. To make full use of the available screen area an appropriate scale factor for the drawing is then selected based upon the overall length of the design. Lengths may range up to 888 feet. Since the CGP-200X requires an origin to be specified for each view shown in a drawing, the location of the origin is also calculated using the overall length and scale factor. This allows the centering of the envelope

graphics on the drawing.

When constructing a geometric model on the CGP-200X, one must first activate a part and a drawing. The drawing used is the Form Part generated by the module MIT.FORMC. Using the layering features of the CGP-200X, the areas of the drawing used to display pressure hull data are blanked, or removed from view. Next, three views of the model are defined. These are top, left, and a standard isometric view. If desired, any of these views may by manually manipulated by the user using the dynamics capability of the Designer System to provide a view of the model from any perspective. The top view is not used in CADSUB in its present form, and is blanked invisible immediately after its creation. It is provided to facilitate arrangement and deck layout. The left view provides a profile of the submarine's port side.

A scale is next drawn to provide orientation to the user. Two scales, one horizontal and one vertical are drawn and labeled. The locations of the aft most point of the forebody and forward most point of the afterbody are indicated with two markers. This entire process is somewhat time consuming and could be eliminated altogether if a single, fixed scale were used. However, clutter is reduced and the overall appearance of the drawing is improved by using a scale which matches the

design.

As explained by Hale, a total of 132 unequally spaced offsets are used to properly portray hull curvature and transition at section junctions. [7] These offsets are now plotted and used to fit a Bezier spline curve which forms the envelope profile. This two dimensional envelope outline is then rotated about the submarine's centerline to form a surface of revolution. A mesh pattern is superimposed on the surface to aid in visualization of the resulting three dimensional form.

The submarine's principal characteristics are next displayed on the drawing, and program execution passes to the next module where the pressure hull is designed.

C. PRESSURE HULL DESIGN

The most complex graphics module now in CADSUB is MIT.PHULL. Information entered interactively by the user is processed to generate both numerical and graphical data on the CRT display.

As in the previous module, the first step taken is to obtain data from pass files. This data includes the diameter, overall length, length of the forebody and afterbody, and the fullness coefficients which define the forebody and afterbody.

The first information required from the user is the

amount of the A weight to be reserved for everbuoyant items such as air flasks, appendages, and structure which are not included in the pressure hull. Following a prompt which displays the A weight, the user inputs the number of tons he wishes to reserve. The required pressure hull volume is then determined by reducing the A weight by the amount to be reserved converting weight to volume.

Next, the isometric view of the outer envelope is blanked and the layers containing the pressure hull design summary and detailed pressure hull segment information are activated. The pressure hull design summary is then initialized by displaying the required and reserved volumes.

The pressure hull is made up of one or more segments, each having fairly simple geometry.[8] The desired shape is selected from a menu. Shapes currently allowed by CADSUB include the following:

Right circular cylinder

Conical Transition

Flat plate end closure

Spherical end closure

Elliptical end closure

Spherical sonar dome with end closure

The user also has the option of requesting the diameter

of the outer envelope at any location aft of the forward perpendicular or changing an existing segment with the edit function.

D. PRESSURE HULL SEGMENTS

Once the shape of the pressure hull segment is selected the user is prompted for information which specifies that shape's geometry. In most cases this consists of the location of the forward and aftmost points of the segment and diameters at these points. All information provided by the user is stored in arrays so that it is readily available if needed.

CADSUB uses this information to generate graphics in a two step process. First a two dimensional profile of the segment is drawn, then rotated about the centerline to form a surface of revolution. A mesh pattern is added to aid visualization.

For simple segments such as cylinders, cones, and flat plate closures this is a line. Spherical segments a formed by rotating a circular arc about the centerline. The graphics command used to draw elliptical segments requires four points. These are two endpoints, one at the outer end of each axis, a point on the ellipse, and the intersection of tangents perpendicular to the axes. These locations are calculated by the program and do not

require any additional input from the user.

Following construction of graphics the volume and center of buoyancy of each segment is calculated analytically. This information, together with the locations and diameters of the ends of the segment are displayed on the drawing as detailed segment information. The total volume used, volume remaining, and center of buoyancy of the entire pressure hull are also updated.

When a spherical end closure is selected the user has the option of directly entering endpoints and diameters or selecting an automatic closure. The automatic closure option requires a specification of a volume to be enclosed by the segment as well as the location and diameter of the base. It then calculates the radius endpoints of the segment.

A spherical sonar dome may be specified with three inputs. These are the sphere diameter, diameter of the access trunk leading to the sonar sphere, and the center of the sphere. Once specified, points defining the sphere in the format required to generate graphics are calculated, the volume and center of gravity of the sphere (without the small volume occupied by the access trunk) are calculated, and the appropriate values are displayed on the drawing. The user is then informed of the location of the aft most point on the sonar sphere

and prompted for the location of the aft most point of the access trunk. A right circular cylinder is used to construct the access trunk.

E. CHANGING THE PRESSURE HULL

As construction of the pressure hull progresses it may become apparent that it is unacceptable that it simply will not balance properly. It is also possible that the user may input an incorrect value and find himself looking at a pressure hull segment which is not what he wanted. Both of these situations may be corrected using the pressure hull editor routine.

When requested, the editor requests the segment to be deleted. All graphics and numerical data relating to this segment are then erased from the screen. Volume used, volume remaining, and pressure hull center of buoyancy are all changed to reflect this deletion.

The user is next presented with a menu similar to the one used to select pressure hull segments. Should the user desire to generate a new segment, he selects the form of the segment and follows prompts as if entering a new segment. Data pertaining to the new segment is displayed in the same location as for the segment which was deleted. Following generation of the new segment the user is returned to the main menu. It is

not necessary to insert a new segment following deletion of another.

F. WHAT'S THE DIAMETER HERE?

The scale generated in the previous module is designed only to give a general idea of location within the envelope. Even with the high quality graphics generated on a Computervision color monitor it is difficult to estimate the diameter of the envelope using only the drawing scale.

It is possible to obtain the diameter of the envelope at any location from the main menu. This function prompts the user for the location at which the diameter is desired. Using the fullness coefficients and lengths of the forebody and afterbody the diameter at that location is calculated. The diameter is then displayed on the CRT and, after a brief pause so that it may be read, the main menu is once again displayed.

Should the user not check the diameter and try to enter a segment larger than the envelope an error message will be displayed. The program will then prompt for new inputs for a segment of the same shape.

G. LEAVING THE PRESSURE HULL

Assuming a pressure hull is finally completed which

has a chance of balancing, the user may select the option to leave this module. He is then asked if a hard copy plot of the drawing is desired, and preparations are made to exit the module.

Prior to entering the next module all data concerning the pressure hull is saved. This data includes all information displayed in the pressure hull summary as well as all detailed segment information.

H. APPENDAGES

Construction of appendages is not now fully implemented in CADSUB. The module MIT.SAIL1 was developed to demonstrate the techniques which may be used to generate a graphic display of appendages.

As in previous modules, pass files are used to obtain information describing the forebody and afterbody. In addition a pass file is used to obtain offsets which define the sail. The user is prompted for the location of the forward edge of the sail. Construction of the sail begins by determining several points along the curve of intersection between the sail and outer envelope. Knowing the location of the sail it is possible to determine the diameter of the envelope at any point along the length of the sail. Then, using the equation describing defining that circle, the exact point of intersection may be determined. The points of

intersection are plotted and connected using a three dimensional Bezier spline curve. This process is done twice, once for the port side and once for the starboard side.

The upper edge of the sail is drawn by plotting offsets at a constant height above the outer envelope. Again, the points are connected with a Bezier spline curve. The upper and lower boundaries of the sail are connected using a ruled surface.

Upon completion of the sail, the user is once again asked if a hard copy plot of the graphic display if desired. Prior to leaving the module the user is also asked if he desires to proceed directly to the balance module (see Hale). If so, the meshes on all three dimensional graphics entities are reduced to a two dimensional form. This is done to enhance the readability of text data generated by the balance module within the pressure hull.

V. EVALUATION OF CADSUB

A. GENERAL COMMENTS

Software developed for this thesis represents a first attempt to incorporate interactive computer graphics into the conceptual design of submarines. The resulting product is not, by any stretch of the imagination, a product which may be marketed as a complete submarine design product. It does, however, demonstrate the utility of linking interactive user control, numerical analysis, and three dimensional graphics into a single powerful design tool.

The interactive nature of the program allows the user to control all important design decisions. Throughout the development of the software it was assumed that any potential user would have a well founded knowledge of the principles of naval architecture and submarine design. It is not the purpose of the software to replace the designer, only to provide a tool which aids in the design process.

It is the coupling of real time graphic displays and numerical analysis which establish the credibility and worth of the computer aided design approach of the software. The many hours required to perform manual calculations and drafting may be reduced significantly

by using computer generated graphics. In addition, a user having even a rudimentary knowledge of the Computervision system can use that ability to go far beyond the capabilities of the software as it now exists.

B. PROPOSED ENHANCEMENTS TO CADSUB

As mentioned previously CADSUB is not now in what could be considered a complete form. While it has a number of desirable features and capabilities, it also has a number of shortcomings and additional features which should be implemented. Many of these are minor and require only a little work to overcome. Others will require a major effort to implement. These areas are outlined below.

1. PROGRAM SPEED AND THE USER INTERFACE

When evaluating the effectiveness of computer software, particularly software which relies heavily on user interaction, the interface between the user and the program is just as important as what calculations the programs performs or what graphics it generates. Much work remains to be done in this area.

One of CADSUB's major defficiencies at this point is the time it requires to complete certain tasks. The most glaring examples involve the insertion of alphanumeric information into a drawing. The creation

and labeling of scales and the updating of pressure hull information appear to be almost fascinating the first time they are viewed by the user. However, computers have the unique ability to slow the apparent passage of time, so that after observing such tasks a couple of times they become only slightly more interesting than watching grass grow. The best solution to this particular problem seems to be translation of the software into a faster language such as Fortran-S.

Improvements can most likely be made in prompts provided to the user for data and decisions. Currently all prompts are provided by printing a menu or a statement on the CRT and waiting for a response to be typed on the keyboard. Computervision designed the CGP-200X as a user friendly system and incorporated a variety of user input devices. Two of these which should be evaluated for inclusion in CADSUB are keyfiles and dynamic menuing.

Keyfiles utilize the digitizer tablet to replace many of the functions of the keyboard. Using keyfiles it is possible to program an area of the digitizer tablet to perform a particular command, and thus eliminate the need to use the keyboard. For example, to select a cylindrical pressure hull segment the user now selects an option from a menu by entering a number on the key-

board. Using keyfiles and the digitizer tablet it would be possible to select the same segment by digitizing an area having a symbolic cylinder drawn on it.

Dynamic menuing is the process of selecting instructions which are displayed as keywords on the CRT screen. As with keyfiles, decisions are entered using the digitizer tablet.

The use of keyfiles or dynamic menus may or may not enhance the utility of CADSUB. They are, however, alternatives which should be investigated.

2. ERROR HANDLING

The only significant error handling routine currently included in the graphics modules of CADSUB is one which verifies that the diameters of pressure hull segments are less than the outer envelope diameter at each end of the segment. Should one select a menu option which is not specified (i.e. option 9 when only 1-7 are allowed) will simply return program execution to the menu for another response.

It is possible to crash the program in a number of ways. For example, entering letters in response to a prompt for numerical information has a catastrophic effect and causes the system to exit the program. A quirk of the Computervision system is that it has considerable difficulty in inserting negative values in a

drawing as text. This problem has been resolved for those areas in which one might expect to find a negative value appear. Should a negative value be encountered which must be displayed elsewhere the results could be disappointing.

3. PROGRAM FLOW

Unless the user is knowledgeable about the Computervision operating system he is locked into the sequence modules are executed by the software. In any case, it is not possible to make changes to a pressure hull design after it has been created and the pressure hull module has been left for the balance module. The ability to re-enter individual modules should be implemented to make it less time consuming to make iterations on a design. In the case of the pressure hull this simply requires an option to activate a previously created design and obtain data from pass files concerning the pressure hull. This data is already stored when the pressure hull module is completed.

C. FEATURES TO BE IMPLEMENTED

The number of features which may be added to the software developed in this thesis is limited only by one's imagination. Development of CADSUB has barely

scratched the surface in an area ripe for research and development.

1. STRUCTURAL DESIGN

No structural design capabilities are now included in CADSUB. This represents a major area in which considerable work may be done, and also offers an opportunity to evaluate the desirability of using finite element analysis in an interactive graphics environment. Computervision supports automatic mesh generation as well as several methods of graphically displaying results.

2. ARRANGEMENTS

One of the most useful applications of computer aided design is likely to be in the development of arrangements. Using solid modeling techniques three dimensional models may be created and maintained throughout the design process.

Development of arrangements should begin with creation of decks. This may be done by either explicitly calculating the lines of intersection of the proposed deck with the pressure hull or by using graphics commands to insert a plane at a specific location. Experimentation is required to select the most efficient method.

Implementation of the capability to zoom in on a

specific area of the design and expand it to use the entire area of the CRT is highly desirable. Along with this capability is the requirement for a dynamic display of location to provide orientation to the user. The scale created by CADSUB when drawing the outer envelope is fixed in place and not suitable for this purpose.

The use of three dimensional solid modeling would be of considerable use in developing three digit weight reports. Each piece of equipment included in the design may be represented by a model stored in a library of parts. Each solid model has a volume, weight, and center of gravity associated with it. This data may be extracted when the model is inserted in the design. Thus, the accuracy of the total weight is dependent on the accuracy of data contained in the library and its completeness. Creation of a library of equipment for use in arrangements is not likely to be a trivial task.

The use of three dimensional modeling may also have a significant impact on reducing interference problems. Automatic interference checking routines would allow the resolution of interferences in the computer model long before actual construction begins.

3. DESIGNS USING NON-CIRCULAR CROSS SECTIONS

This thesis has dealt only with submarine designs having circular cross sections. Pressure hull segments

include only those having a simple shape for which volumes and centers of gravity are easily calculated.

Pressure hull segments which follow the contour or the outer envelope may be easily implemented by adding the appropriate data input prompts to the pressure hull module. A routine to calculate volume and center of buoyancy using numerical integration is also required. The routine to calculate the envelope diameter at any location already exists and may be used to obtain points for integration as well as inclusion in graphics commands.

The design of a submarine not having a circular cross section may be implemented using techniques slightly different from those now used for designs having a circular cross section.

Envelope geometry of designs having a circular cross section are constructed by obtaining offsets, plotting them to obtain a two dimensional representation of the envelope, then rotating the outline about the centerline to obtain a surface of revolution. To construct a model of a non-circular design it is first necessary to plot not individual offsets but a series of cross sections. These cross sections may the be connected using standard graphics commands to generate a three dimensional Bezier spline surface. Pressure hull

segments having non-circular cross sections may be generated in a similar manner.

D. CONCLUSIONS

Computer aided design using both graphics and interactive programming has the potential, if properly managed, to improve the design process by allowing time to examine more alternatives in greater detail. The software developed for this thesis uses only a small fraction of the capability available on a modern CAD system. In fact, much of what is now done in CADSUB can easily be implemented on any one of several microcomputers. However, the potential uses and benefits of computer aided design far exceed those of microcomputers.

The application of CAD technology to the ship design process is just beginning and offers considerable potential. As with all new technology it must be properly managed. CAD should be viewed as a tool to aid the designer in making decisions. It should not be seen as a means to either shorten the design process or to reduce the manpower required to develop a design, but rather as a means to improve the design process.

REFERENCES

1. Krouse, John K., "Automation Revolutionizes Mechanical Design," *High Technology*, March, 1984.

2. Lindgren, Richard K., "Justifying the Cost of CAD," *Computer Graphics World*, September, 1982.

3. *CGOS 200X Operator Guide*, Computervision Corporation, Bedford, Massachusetts, 1983.

4. *CADDS 4X Appearance Reference*, Computervision Corporation, Bedford, Massachusetts, 1983.

5. *CADDS 4X CAD/CAM Concepts Book*, Computervision Corporation, Bedford, Massachusets, 1983.

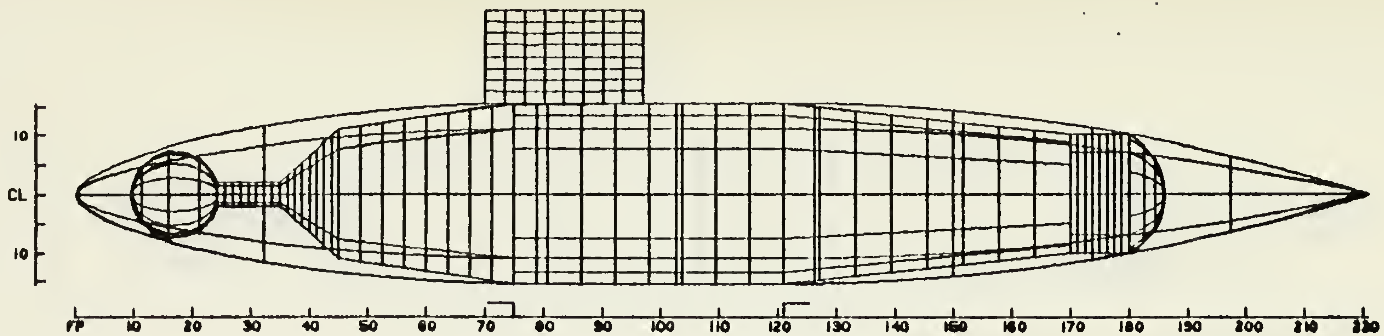
6. *CADDS 4X Menuing Reference*, Computervision Corporation, Bedford, Massachusetts, 1983.

7. Hale, Patrick C., *Computer Aided Conceptual Design of Submarines*, O.E. Thesis, M.I.T., 1984.

8. Jackson, Harry A., *Submarine Design Notes*, Professional Summer at M.I.T., 1983.

APPENDIX I

SAMPLE OUTPUT FROM GRAPHICS MODULES



PRESSURE HULL SEGMENTS

SEG	FWD	AFT	DFWD	DAFT	VOL	LCB
1	9.5	24.2	0	4	1765.4	17
2	24.2	35	4	4	135.4	29.6
3	35	45	4	22	1539.4	42
4	45	75	22	31	16705.4	61.7
5	75	121	31	31	34719.3	98
6	121	170	31	20	25412.6	142
7	170	180	20	20	3141.6	175
8	180	186.2	20	0	1102.4	182.2
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						

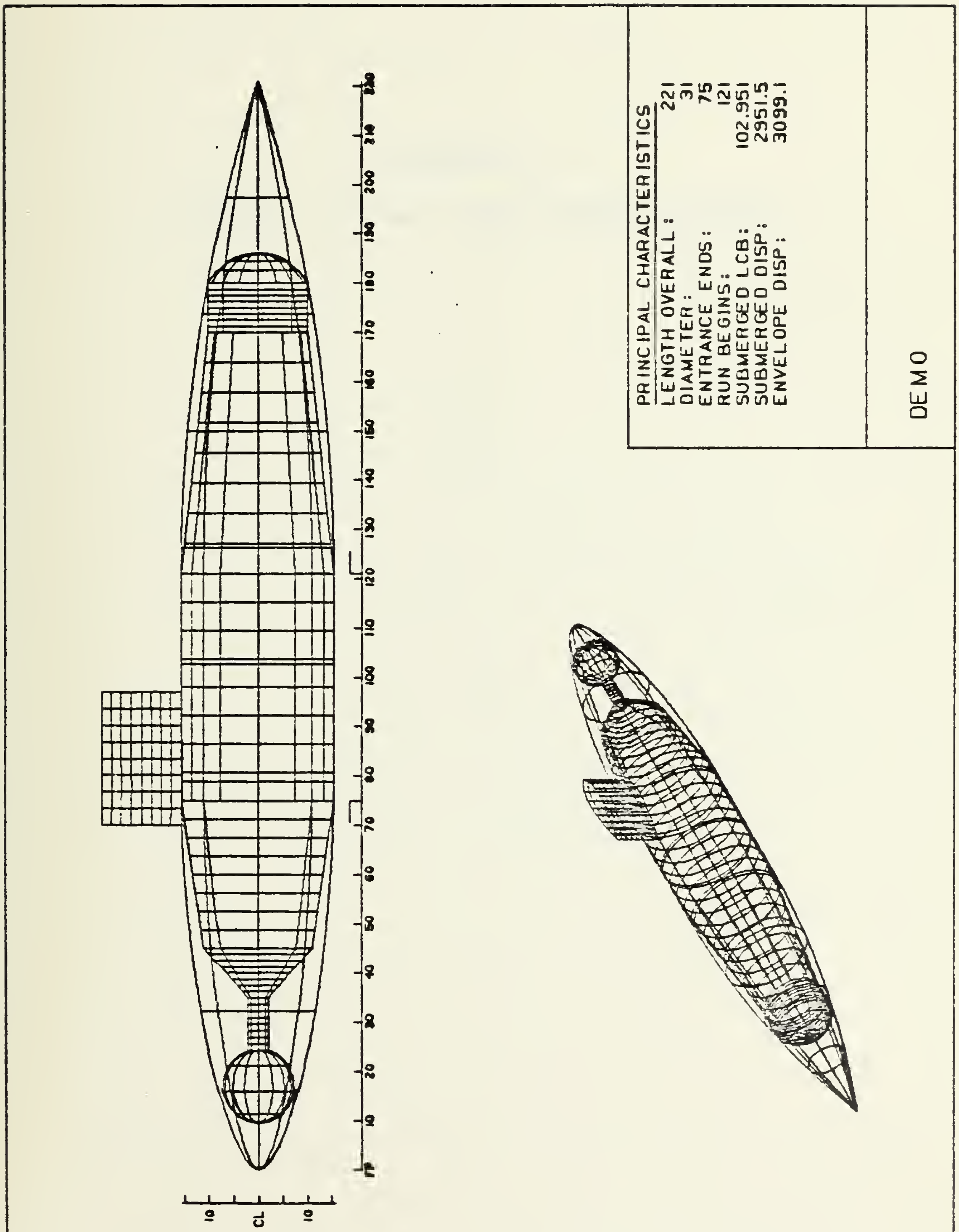
PRESSURE HULL DESIGN SUMMARY

VOLUME REQUIRED:	84521.5
VOLUME USED:	84521.5
VOLUME REMAINING:	0
RESERVED VOLUME:	875
CENTER OF BUOYANCY:	105.2

PRINCIPAL CHARACTERISTICS

LENGTH OVERALL:	221
DIAMETER:	31
ENTRANCE ENDS:	75
RUN BEGINS:	121
SUBMERGED LCB:	102.951
SUBMERGED DISP:	2951.5
ENVELOPE DISP:	3099.1

DEMO


PRINCIPAL CHARACTERISTICS
LENGTH OVERALL: 221
DIAMETER: 31
ENTRANCE ENDS: 75
RUN BEGINS: 121
SUBMERGED LCB: 102.951
SUBMERGED DISP: 2951.5
ENVELOPE DISP: 3099.1
DEMO
CL

APPENDIX II

PROGRAM LISTINGS FOR GRAPHICS MODULES

```
MIT.&BCD.FORMC
 5- 2-84 10:46:13 FUTIL 6.21

<#*******************************************
<# MIT.FORMC
<#*******************************************
<#
<# USED TO SET UP DRAWING FORM
<#
<#*******************************************
ACTIVATE PART MIT.SUB.FORMAT MUNITS FT
ACTIVATE DRAWING FORMC SIZE C
SELECT MODE DRAW
<^U>
INSERT LINE: X.5Y.5,IX21,IY16,IX-21,IY-16<CR>
INSERT LINE: X14Y.5,IY1.5,IX7.5<CR>
<#==============================
<# SET UP INFORMATION TABLES
<#==============================
SELECT LAYER 200
INSERT LINE: X14Y2,X14Y8.75<CR>
INSERT LINE: X.5Y8.75,IX21<CR>
<#------------------------------
<# PRESSURE HULL DESIGN SUMMARY
<#------------------------------
INSERT TEXT "PRESSURE HULL DESIGN SUMMARY" HGT.1875: X14.75Y8.4<CR>
INSERT LINE: X14.75Y8.3,X20.65Y8.3<CR>
INSERT TEXT "VOLUME REQUIRED:  " HGT.1875: X14.75Y8<CR>
INSERT TEXT "VOLUME USED:       " HGT.1875: X14.75Y7.6875<CR>
INSERT TEXT "VOLUME REMAINING:  " HGT.1875: X14.75Y7.375<CR>
INSERT TEXT "RESERVED VOLUME:" HGT.1875: X14.75Y7.0625<CR>
INSERT TEXT "CENTER OF BUOYANCY:" HGT.1875: X14.75Y6.75<CR>
<#------------------------------
<# DETAILED SEGMENT INFORMATION
<#------------------------------
INSERT TEXT "PRESSURE HULL SEGMENTS" HGT.1875: X1Y8.4<CR>
INSERT LINE: X1Y8.3,X5.7Y8.3<CR>
INSERT TEXT "SEG" HGT.1875: X1Y8<CR>
INSERT TEXT "FWD" HGT.1875: X2.75Y8<CR>
INSERT TEXT "AFT" HGT.1875: X4.75Y8<CR>
INSERT TEXT "DFWD" HGT.1875: X6.6875Y8<CR>
INSERT TEXT "DAFT" HGT.1875: X8.6875Y8<CR>
INSERT TEXT "VOL" HGT.1875: X10.75Y8<CR>
INSERT TEXT "LCB" HGT.1875: X12.75Y8<CR>
SEG=1
XSEG=1.5
YSEG=7.6875
REPEAT
  &SEG=SEG
  INSERT TEXT [&SEG] HGT.1875 RJT: X[XSEG]Y[YSEG]<CR>
  SEG=SEG+1
  YSEG=YSEG-.3125
UNTIL (SEG.GT.20)
<#------------------------------
<# PRINCIPAL CHARACTERISTICS
<#------------------------------
SELECT LAYER 199
INSERT LINE: X14Y2,IY4,IX7.5
INSERT TEXT "PRINCIPAL CHARACTERISTICS" HGT.1875: X14.75Y5.65<CR>
```

```
57!INSERT LINE: X14.75Y5.55,IX5.1<CR>
58!INSERT TEXT "LENGTH OVERALL:       " HGT.1875: X14.75Y5.25<CR>
59!INSERT TEXT "DIAMETER:             " HGT.1875: X14.75Y4.9375<CR>
60!INSERT TEXT "ENTRANCE ENDS:        " HGT.1875: X14.75Y4.625<CR>
61!INSERT TEXT "RUN BEGINS:           " HGT.1875: X14.75Y4.3125<CR>
62!INSERT TEXT "SUBMERGED LCB:        " HGT.1875: X14.75Y4<CR>
63!INSERT TEXT "SUBMERGED DISP:" HGT.1875: X14.75Y3.6875<CR>
64!INSERT TEXT "ENVELOPE DISP:" HGT.1875: X14.75Y3.375<CR>
65!SELECT LAYER 205
66!INSERT TEXT "-" HGT.1875: X20.28Y7.375<CR>
67!<^U>
68!EXIT PAR F
69!PREP NFIG MIT.SUB.FORMAT DRAW FORMC
70!END
```

MIT.&BCD.ENV
5-15-84 12:26:43 FUTIL 6.21

```
<#*********************************
<# MIT.ENV
<#*********************************
<#
<# -GENERATES VIEWS
<# -GENERATES DRAWING OF ENVELOPE
<#
<#*********************************
DIM X(132),Y(132)
DEC=0
EXECV
#COLOR
PRNT
PRNT
PRNT
READ (IS THIS A C>OLOR OR M>ONOCHROME TERMINAL? )&TYPE
IF (&TYPE.NE."C".AND.&TYPE.NE."M") GOTO COLOR
PRNT
PRNT
PRNT
PRNT OBTAINING OFFSETS FROM DATA FILE...
OPENR 1,"MIT.POINTS"
<^U>
<#*********************************
<# READ OFFSETS
<#*********************************
I=1
REPEAT
  READF 1,&TXT
  X(I)=&TXT
  READF 1,&TXT
  Y(I)=&TXT
  I=I+1
UNTIL (I.EQ.133)
<#*********************************
<# READ PRINCIPAL DIMENSIONS
<#*********************************
OPENR 2,"MIT.PASSPH1"
READF 2,&DIAM
READF 2,&LOA
READF 2,&LF
READF 2,&LLA
READF 2,&LCBSB
READF 2,&DSUB
READF 2,&DENV
DIAM=&DIAM
LOA=&LOA
LF=&LF
LLA=&LLA
LCBSUB=&LCBSB
DSUB=&DSUB
DENV=&DENV
LA=LOA-LLA
&LA=LA
OPENR 3,"MIT.PASSPH2"
READF 3,&AWGHT
```

```
READF 3,&NAME
<#------------------------------
<# ROUND OFF VALUES
<#------------------------------
DIAM=AINT((DIAM+.05)*10)/10
&DIAM=DIAM
LOA=(AINT((LOA+.05)*10))/10
&LOA=LOA
LF=(AINT((LF+.05)*10))/10
&LF=LF
LA=(AINT((LA+.05)*10))/10
&LA=LA
&LCBSB=LCBSUB
DSUB=(AINT((DSUB+.05)*10))/10
&DSUB=DSUB
DENV=(AINT((DENV+.05)*10))/10
&DENV=DENV
<#*********************************************
<# SET UP CSIZE DRAWING
<#*********************************************
#CSIZE
<#--------------------------------------------
<# SELECT SCALE FACTOR
<#--------------------------------------------
IF (LOA.LE.148) SCL=8
IF (LOA.GT.148.AND.LOA.LE.185) SCL=10
IF (LOA.GT.185.AND.LOA.LE.222) SCL=12
IF (LOA.GT.222.AND.LOA.LE.296) SCL=16
IF (LOA.GT.296.AND.LOA.LE.370) SCL=20
IF (LOA.GT.370.AND.LOA.LE.440) SCL=24
IF (LOA.GT.440.AND.LOA.LE.592) SCL=32
IF (LOA.GT.592.AND.LOA.LE.740) SCL=40
IF (LOA.GT.740) SCL=48
<#------------------------------
<# DETERMINE LOCATION OF ORIGIN
<#------------------------------
XORG=1.75+(18.5-LOA/SCL)/2
YORG=12.25
<#------------------------------
<# ACTIVATE DRAWING
<#------------------------------
ACTIVATE PART MIT.SUBDRA MUNITS FT
ACTIVATE DRAWING FORMAT FORM MIT.SUB.FORMAT DRAW FORMC
SELECT LDISCRIMINATION WHITE LAYER 0 <CR>
IF (&TYPE.EQ."C") DISCRIMINATE LAYER
SELECT LDISCRIMINATION YELLOW LAYER 1.5 <CR>
IF (&TYPE.EQ."C") DISCRIMINATE LAYER
SELECT LDISCRIMINATION RED LAYER 2 <CR>
IF (&TYPE.EQ."C") DISCRIMINATE LAYER
SELECT MODE DRAW
ECHO LAYER EXCLUDE 200
ECHO LAYER EXCLUDE 205
SELECT LAYER 0
<#------------------------------
<# DEFINE VIEWS
<#------------------------------
DEF VIEW TOP CPL TOP SCALE 1 IN TO [SCL] FT TILT 90: X[XORG]Y[YORG]<*
,X1Y8.5,X21Y16<CR>
BLANK VIEW NAME 'TOP'
DEF VIEW LEFT CPL LEFT SCALE 1 IN TO [SCL] FT: X[XORG]Y[YORG]<*
```

```
.X1Y8.5,X21Y16<CR>
DEF VIEW ISO CPL ISO SCALE 1 IN TO [SCL+/] FT: X11Y6.9<*
.X1Y1,X13Y8<CR>
SEL CPL LEFT
<#-------------------------------
<# DRAW SCALE
<#-------------------------------
IF (DEC.EQ.1) GOTO ENV
<#*********************************************
<# GENERATE SCALE
<#*********************************************
#SCALE
<#-------------------------------
<# INITIALIZE VARIABLES
<#-------------------------------
SCRAD=DIAM/(SCL*2)
YSC=11.8-SCRAD
SCLOA=LOA/SCL
INC=10/SCL
YINC=YSC+.125
<#-------------------------------
<# DRAW HORIZONTAL SCALE LINE
<#-------------------------------
INSERT LINE: X[XORG]Y[YSC],X[XORG+SCLOA]Y[YSC]
DISC ENT CLEAR RED
<#-------------------------------
<# DRAW SCALE MARKINGS
<#-------------------------------
XINC=XORG
REPEAT
  INS LIN: X[XINC]Y[YINC],X[XINC]Y[YSC]
  XINC=XINC+INC
  XDIF=XINC-XORG
UNTIL (XDIF.GT.SCLOA)
<#-------------------------------
<# LABEL SCALE
<#-------------------------------
ILBL=10
XINC=XORG
XOFF=.1
YLBL=YSC-.2
INS TEXT "FP" HGT.125: X[XORG-.125]Y[YLBL]<CR>
<#-------------------------------
<# DETERMINE LABEL INCREMENT
<#-------------------------------
IF (SCL.LT.24) GOTO DRLBL
#LBL2
ILBL=20
INC=2*INC
GOTO DRLBL
#DRLBL
XINC=XINC+INC
LBL=ILBL
&LBL=ILBL
REPEAT
  IF (LBL.GE.100) XOFF=.130
  XLBL=XINC-XOFF
  INS TEXT [&LBL] HGT.125: X[XLBL]Y[YLBL]<CR>
  LBL=LBL+ILBL
  &LBL=LBL
```

```
   XINC=XINC+INC
   XDIF=XINC-XORG
UNTIL (XDIF.GT.SCLOA)
<#-------------------------------
<# DRAW ENTRANCE AND RUN MARKERS
<#-------------------------------
SCLF=LF/SCL
SCLA=LA/SCL
XLF=XORG+SCLF
XLA=XORG+SCLA
INSERT LINE: X[XLF]Y[YSC],X[XLF]Y[YSC+.1875],X[XLF-.375]<CR>
INSERT LINE: X[XLA]Y[YSC],X[XLA]Y[YSC+.1875],X[XLA+.375]<CR>
<#-------------------------------
<# DRAW VERTICAL SCALE
<#-------------------------------
XVSC=XORG-.575
INSERT LINE: X[XVSC]Y[YORG-SCRAD],X[XVSC]Y[YORG+SCRAD]<CR>
<#-------------------------------
<# DRAW SCALE MARKINGS
<#-------------------------------
YINC=YORG
INC=5/SCL
REPEAT
   INSERT LINE: X[XVSC]Y[YINC],IX.125<CR>
   YINC=YINC+INC
UNTIL (YINC.GT.YORG+SCRAD)
YINC=YORG
REPEAT
   INSERT LINE: X[XVSC]Y[YINC],IX.125<CR>
   YINC=YINC-INC
UNTIL (YINC.LT.YORG-SCRAD)
<#-------------------------------
<# LABEL VERTICAL SCALE
<#-------------------------------
XLBL=XVSC-.075
YINC=YORG-.0625
INC=10/SCL
INSERT TEXT "CL" HGT.125 RJT: X[XLBL]Y[YINC]<CR>
ILBL=10
IF(SCL.LT.24) GOTO DRVLB
ILBL=20
INC=2*INC
#DRVLB
YLBL=YINC+INC
LBL=ILBL
&LBL=LBL
REPEAT
   INSERT TEXT [&LBL] HGT.125 RJT: X[XLBL]Y[YLBL]<CR>
   YLBL=YLBL+INC
   LBL=LBL+ILBL
   &LBL=LBL
UNTIL (YLBL.GT.YORG+SCRAD)
YLBL=YINC-INC
LBL=ILBL
&LBL=ILBL
REPEAT
   INSERT TEXT [&LBL] HGT.125 RJT: X[XLBL]Y[YLBL]<CR>
   YLBL=YLBL-INC
   LBL=LBL+ILBL
   &LBL=LBL
```

```
UNTIL (YLBL.LT.YORG-SCRAD-.0625)
<#-----------------------------------
<# DRAW ENVELOPE
<#-----------------------------------
#ENV
SELECT MODE MODEL
SELECT LAYER 1
INS LIN: X0Y0.X[LOA]Y0
I=1
TG=1
&TG=TG
INSERT BSPLINE INTR TAG=[&TG]:<#
#NEXT X[X(I)]Y[Y(I)],<#
I=I+1
IF (I.LE.132) GOTO NEXT
<CR>
INSERT SREV HIANG 360 MESH 8X8: TAG [&TG];X0Y0.X[LOA]Y0<CR>
<#-----------------------------------
<# DISPLAY PRINCIPAL CHARACTERISTICS
<#-----------------------------------
SELECT MODE DRAW
SELECT LAYER 0
INSERT TEXT [&NAME] HGT.25: X14.5Y1<CR>
SELECT LAYER 199
INSERT TEXT [&LOA] HGT.1875 RJT: X20.25Y5.25<CR>
INSERT TEXT [&DIAM] HGT.1875 RJT: X20.25Y4.9375<CR>
INSERT TEXT [&LF] HGT.1875 RJT: X20.25Y4.625<CR>
INSERT TEXT [&LA] HGT.1875 RJT: X20.25Y4.3125<CR>
INSERT TEXT [&LCBSB] HGT.1875 RJT: X20.25Y4<CR>
INSERT TEXT [&DSUB] HGT.1875 RJT: X20.25Y3.6875<CR>
INSERT TEXT [&DENV] HGT.1875 RJT: X20.25Y3.375<CR>
<^U>
#END
RUN NEW MIT.PHULL
END
```

```
<#*******************************************
<# MIT.PHULL
<#*******************************************
<#
<# USED TO GENERATE PRESSURE HULL GEOMETRY
<#    -DRAWS 3D REPRESENTATION OF HULL FORM
<#    -CALCULATES AND DISPLAYS PARAMETERS
<#
<#*******************************************
DIM AFT(20),FWD(20),PDA(20),PDF(20),LEN(20),VOL(20),XCB(20)
DIM TPE(20)
PI=3.14159265
<#*********************************
<# OBTAIN DATA FROM PASS FILES
<#*********************************
OPENR 1,"MIT.PASSPH1"
READF 1,&TYPE
DIAM=&TYPE
READF 1,&TYPE
LOA=&TYPE
READF 1,&TYPE
LF=&TYPE
READF 1,&TYPE
LA=&TYPE
READF 1,&TYPE
READF 1,&TYPE
READF 1,&TYPE
READF 1,&TYPE
NF=&TYPE
READF 1,&TYPE
NA=&TYPE
OPENR 2,"MIT.PASSPH2"
READF 2,&TYPE
AWGHT=&TYPE
RAD=DIAM/2
EXECV
PRNT
PRNT
PRNT
&TYPE=AWGHT
PRNT AWEIGHT=[&TYPE]
AWGHT=(AINT((AWGHT+.05)*10))/10
READ (NUMBER OF TONS OF AWEIGHT TO BE RESERVED IS —> ) &TYPE
RES=&TYPE
TWGHT=AWGHT-RES
TVOL=TWGHT*35
RES=RES*35
&RES=RES
<#----------------------------
<# UPDATE SUMMARY
<#----------------------------
<^U>
SELECT MODE DRAW
BLANK VIEW NAME 'ISO'
ECHO LAYER INCLUDE 200
SELECT LAYER 200
```

```
&TVOL=TVOL
INSERT TEXT [TVOL] HGT.1875 RJT: X20.25Y8<CR>
INSERT TEXT "0" HGT.1875 RJT: X20.25Y7.0875<CR>
INSERT TEXT [TVOL] HGT.1875 RJT: X20.25Y7.375<CR>
INSERT TEXT [RES] HGT.1875 RJT: X20.25Y7.0725<CR>
INSERT TEXT "0" HGT.1875 RJT: X20.25Y6.75<CR>
<#-----------------------------
<# SELECT LAYER AND TAG
<#-----------------------------
SELECT MODE MODEL
SELECT CPL LEFT
SELECT LAYER 2 <CR>
<^U>
TG=11
#SEG
ERR1=0
ERR2=0
&TG=TG
ITG=TG-10
<#********************************
<# MAIN MENU
<#********************************
EXECV
#MENU1
PRNT
PRNT SELECT FORM OF PRESSURE HULL SEGMENT
PRNT 1-CYLINDER              2-CONICAL SEGMENT     3-FLAT PLATE CLOSURE
PRNT 4-SPHERICAL CLOSURE 5-ELLIPTICAL CLOSURE 6-SONAR AND ACCESS
READ (7-OBTAIN DIAMETER    E-EDIT                 X-LEAVE         --->) &TYPE
IF (&TYPE.EQ."1") GOSUB CYLIN
IF (&TYPE.EQ."2") GOSUB CONE
IF (&TYPE.EQ."3") GOSUB PLATE
IF (&TYPE.EQ."4") GOSUB SPH1
IF (&TYPE.EQ."5") GOSUB ELLIP
IF (&TYPE.EQ."6") GOSUB SONAR
IF (&TYPE.EQ."7") GOSUB GETR
IF (&TYPE.EQ."X") GOSUB'STP
IF (&TYPE.EQ."E") GOSUB EDIT
<#
IF (&TYPE.EQ."7") GOTO MENU1
IF (&TYPE.EQ."R") GOTO SEG1
TPE(ITG)=&TYPE
GOSUB UPDATE
#SEG1
TG=TG+1
GOTO SEG
#STP
GOTO END
<#********************************
<# SPHERICAL SEGMENT OF ONE BASE
<#********************************
#SPH1
PRNT
PRNT
PRNT SPHERICAL ENDCLOSURE
PRNT CHOOSE METHOD OF INPUT:
READ (1-INPUT ENDPOINTS 2-CALCULATE ENDPOINTS FROM VOLUME---> ) &TYPE
IF (&TYPE.EQ."2") GOTO CALPT
IF (&TYPE.NE."1") GOTO SPH1
EXECV
```

```
#SPI
PRNT INPUT ENDPOINTS
READ (FORWARD MOST POINT OF SEGMENT IS --> ) &FWD
READ (AFT MOST POINT OF SEGMENT IS      --> ) &AFT
READ (DIAMETER OF FORWARD MOST POINT IS--> ) &PDF
READ (DIAMETER OF AFT MOST POINT IS     --> ) &PDA
FWD(ITG)=&FWD
AFT(ITG)=&AFT
PDF(ITG)=&PDF
PDA(ITG)=&PDA
LEN(ITG)=AFT(ITG)-FWD(ITG)
GOSUB CHECK
IF (ERR1.EQ.1.OR.ERR2.EQ.1) GOTO SPI
GOSUB SPSET
GOTO SPDRA
<#--------------------------------------
<# CALCULATE POINTS FROM GIVEN VOLUME
<#--------------------------------------
#CALPT
EXECV
PRNT CALCULATE ENDPOINTS FROM VOLUME
READ (REQUIRED VOLUME OF SEGMENT IS --> ) &VOL
READ (LOCATION OF SEGMENT BASE IS    --> ) &LOC
READ (DIAMETER OF SEGMENT BASE IS    --> ) &PDF
VOLUME=&VOL
LOC=&LOC
<#------------------------------
<# DETERMINE ORIENTATION
<#------------------------------
IF (LOC.LT.(LOA/2)) PDF(ITG)=0
IF (LOC.LT.(LOA/2)) PDA(ITG)=&PDF
IF (LOC.GT.(LOA/2)) PDA(ITG)=0
IF (LOC.GT.(LOA/2)) PDF(ITG)=&PDF
<#---------------------------
<# CALCULATE LENGTH OF SEGMENT
<#---------------------------
IF (PDF(ITG).EQ.0) R=PDA(ITG)/2
IF (PDA(ITG).EQ.0) R=PDF(ITG)/2
A=3*(R**2)
B=6*VOLUME/PI
SQR=SQRT((B**2)/4+(A**3)/27)
AA=(B/2+SQR)**(1/3)
BA=B/2-SQR
IF (BA.LT.0) BC=-BA
IF (BA.LT.0) BB=-(BC**(1/3))
IF (BA.GT.0) BB=BA**(1/3)
LEN(ITG)=AA+BB
IF (PDF(ITG).EQ.0) FWD(ITG)=LOC-LEN(ITG)
IF (PDF(ITG).EQ.0) AFT(ITG)=LOC
IF (PDA(ITG).EQ.0) FWD(ITG)=LOC
IF (PDA(ITG).EQ.0) AFT(ITG)=LOC+LEN(ITG)
GOSUB CHECK
IF (ERR1.EQ.1.OR.ERR2.EQ.2) GOTO CALPT
GOSUB SPSET
GOTO SPDRA
<#------------------------------
<# DRAW SEGMENT
<#------------------------------
#SPDRA
<^U>
```

```
INSERT ARC TAG=[&TG]: X[X1]Y[Y1],X[X2]Y0,X[X1]Y[-Y1]<CR>
INSERT SREV HIANG=180 MESH 8X8 : TAG[&TG];X0Y0,X[LOA]Y0<CR>
<^U>
<#------------------------------
<#CALCULATE PROPERTIES
<#------------------------------
SRAD=(PD**2)/(8*LEN(ITG))+LEN(ITG)/2
VOL(ITG)=(LEN(ITG)**2)*PI*(3*SRAD-LEN(ITG))/3
CB=SRAD-.75*((2*SRAD-LEN(ITG))**2)/(3*SRAD-LEN(ITG))
IF (PDF(ITG).EQ.0) XCB(ITG)=FWD(ITG)+CB
IF (PDA(ITG).EQ.0) XCB(ITG)=AFT(ITG)-CB
RTNSUB
<#--------------------------------
<# SET UP COORDINATES FOR PLOTTING
<#--------------------------------
#SPSET
IF (PDF(ITG).EQ.0) PD=PDA(ITG)
IF (PDF(ITG).EQ.0) X1=AFT(ITG)
IF (PDF(ITG).EQ.0) X2=FWD(ITG)
IF (PDF(ITG).EQ.0) Y1=PDA(ITG)/2
IF (PDA(ITG).EQ.0) PD=PDF(ITG)
IF (PDA(ITG).EQ.0) X1=FWD(ITG)
IF (PDA(ITG).EQ.0) X2=AFT(ITG)
IF (PDA(ITG).EQ.0) Y1=-PDF(ITG)/2
RTNSUB
<#*******************************
<# CONICAL SEGMENT
<#*******************************
#CONE
PRNT
PRNT
PRNT CONICAL SEGMENT
READ (FORWARD MOST POINT OF SEGMENT IS --> ) &FWD
READ (AFT MOST POINT OF SEGMENT IS      --> ) &AFT
READ (DIAMETER AT FORWARD MOST POINT IS--> ) &PDF
READ (DIAMETER AT AFT MOST POINT IS     --> ) &PDA
FWD(ITG)=&FWD
AFT(ITG)=&AFT
PDF(ITG)=&PDF
PDA(ITG)=&PDA
LEN(ITG)=AFT(ITG)-FWD(ITG)
GOSUB CHECK
IF (ERR1.EQ.1.OR.ERR2.EQ.1) GOTO CONE
<#------------------------------
<# DRAW CONICAL SEGMENT
<#------------------------------
X1=FWD(ITG)
X2=AFT(ITG)
Y1=PDF(ITG)/2
Y2=PDA(ITG)/2
<^U>
INSERT LINE TAG=[&TG]: X[X1]Y[Y1],X[X2]Y[Y2]<CR>
INSERT SREV HIANG=360 MESH 8X8 : TAG[TG];X0Y0,X[LOA]Y0<CR>
<^U>
<#----------------------------
<# CALCULATE CONICAL PROPERTIES
<#----------------------------
VOL(ITG)=LEN(ITG)*PI*(PDF(ITG)**2+PDA(ITG)**2+PDF(ITG)*PDA(ITG))/12
IF (PDA(ITG).LT.PDF(ITG)) GOTO CONB
D1=PDA(ITG)
```

```
D2=PDF(ITG)
GOTO CONC
#CONB
D1=PDF(ITG)
D2=PDA(ITG)
#CONC
CB=LEN(ITG)*(D1**2+2*D1*D2+3*D2**2)/((D1**2+D1*D2+D2**2)*4)
IF (PDA(ITG).GT.PDF(ITG)) XCB(ITG)=AFT(ITG)-CB
IF (PDF(ITG).GE.PDA(ITG)) XCB(ITG)=FWD(ITG)+CB
RTNSUB
<#********************************
<# CYLINDRICAL SEGMENT
<#********************************
#CYLIN
PRNT
PRNT
PRNT CYLINDRICAL SEGMENT
READ (FORWARD MOST POINT OF SEGMENT IS   --> ) &FWD
READ (AFT MOST POINT  OF THIS SEGMENT IS--> ) &AFT
READ (DIAMETER OF CYLINDRICAL SEGMENT IS--> ) &PDA
FWD(ITG)=&FWD
AFT(ITG)=&AFT
PDF(ITG)=&PDA
PDA(ITG)=PDF(ITG)
LEN(ITG)=AFT(ITG)-FWD(ITG)
GOSUB CHECK
IF (ERR1.EQ.1.OR.ERR2.EQ.1) GOTO CYLIN
<#------------------------------
<# DRAW CYLINDER
<#------------------------------
#DRCYL
<^U>
INS LIN TAG=[&TG]: X[FWD(ITG)]Y[PDF(ITG)/2]<#
.X[AFT(ITG)]Y[PDA(ITG)/2]<CR>
INSERT SREV HIANG 360 MESH 8X16 : TAG[&TG]:X0Y0.X[LOA]Y0<CR>
<^U>
<#------------------------------
<# CALCULATE PROPERTIES
<#------------------------------
VOL(ITG)=LEN(ITG)*PI*(PDF(ITG)**2)/4
XCB(ITG)=FWD(ITG)+LEN(ITG)/2
RTNSUB
<#********************************
<# ELLIPTICAL END CLOSURE
<#********************************
#ELLIP
PRNT
PRNT
PRNT ELLIPTICAL END CLOSURE
READ (FORWARD MOST POINT OF SEGMENT IS --> ) &FWD
READ (AFT MOST POINT OF SEGMENT IS     --> ) &AFT
READ (DIAMETER OF FORWARD MOST POINT IS--> ) &PDF
READ (DIAMETER OF AFT MOST POINT IS    --> ) &PDA
FWD(ITG)=&FWD
AFT(ITG)=&AFT
PDF(ITG)=&PDF
PDA(ITG)=&PDA
LEN(ITG)=AFT(ITG)-FWD(ITG)
GOSUB CHECK
IF (ERR1.EQ.1.OR.ERR2.EQ.1) GOTO ELLIP
```

```
<#--------------------------------
<# SET UP POINTS FOR PLOTTING
<#--------------------------------
IF (PDF(ITG).EQ.0) X1=FWD(ITG)
IF (PDF(ITG).EQ.0) X2=FWD(ITG)
IF (PDF(ITG).EQ.0) Y2=PDA(ITG)/2
IF (PDF(ITG).EQ.0) X3=AFT(ITG)
IF (PDA(ITG).EQ.0) X1=AFT(ITG)
IF (PDA(ITG).EQ.0) X2=AFT(ITG)
IF (PDA(ITG).EQ.0) Y2=PDF(ITG)/2
IF (PDA(ITG).EQ.0) X3=FWD(ITG)
Y3=Y2
X4=(X1+X3)/2
IF (PDF(ITG).EQ.0) XX=X4-X1
IF (PDA(ITG).EQ.0) XX=X1-X4
Y4=SQRT((Y2**2)*(1-(XX**2)/(LEN(ITG)**2)))
<#--------------------------------
<# DRAW ELLIPTICAL END CLOSURE
<#--------------------------------
<^U>
INSERT CONIC TAG=[&TG]: X[X1]Y0,X[X2]Y[Y2],X[X3]Y[Y3],X[X4]Y[Y4]<CR>
INSERT SREV HIANG=360 MESH=8X8 : TAG[&TG]:X0Y0,X[LOA]Y0<CR>
<^U>
<#-----------------------------------
<# CALCULATE ELLIPTICAL PROPERTIES
<#-----------------------------------
VOL(ITG)=(PI/6)*LEN(ITG)*((2*Y2))**2
CB=.375*LEN(ITG)
IF (PDF(ITG).EQ.0) XCB(ITG)=AFT(ITG)-CB
IF (PDA(ITG).EQ.0) XCB(ITG)=FWD(ITG)+CB
RTNSUB
<#*********************************
<# FLAT PLATE CLOSURE
<#*********************************
#PLATE
PRNT
PRNT
PRNT FLAT PLATE END CLOSURE
READ (LOCATION OF FLAT PLATE CLOSURE IS --> ) &FWD
READ (DIAMETER OF FLAT PLATE CLOSURE IS --> ) &PDF
FWD(ITG)=&FWD
AFT(ITG)=&FWD
PDF(ITG)=&PDF
PDA(ITG)=&PDF
LEN(ITG)=0
VOL(ITG)=0
XCB(ITG)=&FWD
GOSUB CHECK
IF (ERR1.EQ.1.OR.ERR2.EQ.1) GOTO PLATE
<#--------------------------------
<# DRAW FLAT PLATE END CLOSURE
<#--------------------------------
X1=FWD(ITG)
Y2=PDF(ITG)/2
<^U>
INSERT LINE TAG=[&TG]: X[X1]Y0,X[X1]Y[Y2]<CR>
INSERT SREV HIANG=360 MESH 8X8: TAG[TG]:X0Y0,X[LOA]Y0<CR>
<^U>
RTNSUB
<#*********************************
```

```
<# SONAR DOME ROUTINE
<#**********************************
#SONAR
PRNT
PRNT
PRNT
PRNT SONAR SPHERE
READ (DIAMETER OF SONAR SPHERE --> ) &SDIAM
READ (DIAMETER OF ACCESS TRUNK --> ) &PDA
READ (CENTER OF SONAR SPHERE    --> ) &CEN
SDIAM=&SDIAM
DTRNK=&PDA
CEN=&CEN
SRAD=SDIAM/2
SD=.5*SQRT(SDIAM**2-DTRNK**2)
SH=SRAD-SD
XL=CEN
<#------------------------------
<# SET UP POINTS AND PLOT
<#------------------------------
X1=CEN+SD
X2=CEN-SRAD
Y1=DTRNK/2
FWD(ITG)=X2
AFT(ITG)=X1
PDF(ITG)=0
PDA(ITG)=&PDA
LEN(ITG)=AFT(ITG)-FWD(ITG)
<^U>
INSERT ARC TAG=[&TG]: X[X1]Y[Y1],X[X2]Y0,X[X1]Y[-Y1]<CR>
INSERT SREV HIANG=180 MESH8X8 : TAG[&TG]:X0Y0,X[LOA]Y0<CR>
<^U>
<#------------------------------
<# CALCULATE PROPERTIES
<#------------------------------
VSPH=4*PI*(SRAD**3)/3
VSEG=(PI/3)*(SH**2)*(3*SRAD-SH)
VOL(ITG)=VSPH-VSEG
CBSEG=CEN+.75*((2*SRAD-LEN(ITG))**2)/(3*SRAD-LEN(ITG))
XCB(ITG)=(VSPH*CEN-VSEG*CBSEG)/VOL(ITG)
IF (DTRNK.EQ.0) RTNSUB
IF (DTRNK.NE.0) GOSUB UPDATE
<#------------------------------
<# CONSTRUCT ACCESS TRUNK
<#------------------------------
&AFT=AFT(ITG)
EXECV
PRNT
PRNT
PRNT THE AFT MOST POINT OF THE SONAR SPHERE IS LOCATED AT [&AFT]
READ ( AFT MOST END OF THE ACCESS TRUNK WILL BE LOCATED AT --> ) &AFT
TG=TG+1
&TG=TG
ITG=TG-10
TPE(ITG)=3
FWD(ITG)=X1
AFT(ITG)=&AFT
PDF(ITG)=PDA(ITG-1)
PDA(ITG)=PDF(ITG)
LEN(ITG)=AFT(ITG)-FWD(ITG)
```

```
GOSUB DRCYL
RTNSUB
<#***************************************
<# UPDATE STATUS AND DISPLAY VARIABLES
<#***************************************
#UPDATE
<^U>
SELECT MODE DRAW
SELECT LAYER 200
<#------------------------------
<# UPDATE SEGMENT SUMMARY
<#------------------------------
YLOC=8-.3125*ITG
<#------------------------------
<# ROUND OFF NUMBERS
<#------------------------------
FWD(ITG)=(AINT((FWD(ITG)+.05)*10))/10
AFT(ITG)=(AINT((AFT(ITG)+.05)*10))/10
PDF(ITG)=(AINT((PDF(ITG)+.05)*10))/10
PDA(ITG)=(AINT((PDA(ITG)+.05)*10))/10
VOL(ITG)=(AINT((VOL(ITG)+.05)*10))/10
XCB(ITG)=(AINT((XCB(ITG)+.05)*10))/10
&FWD=FWD(ITG)
&AFT=AFT(ITG)
&DFWD=PDF(ITG)
&DAFT=PDA(ITG)
&VOL=VOL(ITG)
&XCB=XCB(ITG)
INSERT TEXT [&FWD] HGT.1875 RJT: X3.30Y[YLOC]<CR>
INSERT TEXT [&AFT] HGT.1875 RJT: X5.30Y[YLOC]<CR>
INSERT TEXT [&DFWD] HGT.1875 RJT: X7.30Y[YLOC]<CR>
INSERT TEXT [&DAFT] HGT.1875 RJT: X9.30Y[YLOC]<CR>
INSERT TEXT [&VOL] HGT.1875 RJT: X11.5Y[YLOC]<CR>
INSERT TEXT [&XCB] HGT.1875 RJT: X13.3Y[YLOC]<CR>
<#------------------------------
<# UPDATE DESIGN SUMMARY
<#------------------------------
PVOL=PVOL+(AINT(VOL(ITG)*10))/10
RVOL=TVOL-PVOL
IF (RVOL.LT.0) NFLG=1
IF (NFLG.EQ.1) RVOL=ABS(RVOL)
&PVOL=PVOL
&RVOL=RVOL
MOMENT=MOMENT+VOL(ITG)*XCB(ITG)
LCB=MOMENT/PVOL
LCB=(AINT((LCB+.05)*10))/10
IF (TPE(ITG).EQ.5) RTNSUB
&LCB=LCB
DELETE ENTITY: X20.25Y7.6875<CR>
INSERT TEXT [&PVOL] HGT.1875 RJT: X20.25Y7.6875<CR>
DELETE ENTITY: X20.25Y7.375<CR>
INSERT TEXT [&RVOL] HGT.1875 RJT: X20.25Y7.375<CR>
IF (NFLG.EQ.1) ECHO LAYER INCLUDE 205
IF (NFLG.EQ.0) ECHO LAYER EXCLUDE 205
DELETE ENTITY: X20.25Y6.75<CR>
INSERT TEXT [&LCB] HGT.1875 RJT: X20.25Y6.75<CR>
IF (NFLG.EQ.1) RVOL=-RVOL
IF (NFLG.EQ.1) NFLG=0
SELECT MODE MODEL
SELECT LAYER 2
```

```
<^U>
RTNSUB
<#********************************
<# PRESSURE HULL EDIT SUBROUTINE
<#********************************
#EDIT
<#----------------------------
<# DELETE PRESSURE HULL SEGMENT
<#----------------------------
PRNT
PRNT
PRNT
READ (SEGMENT TO BE DELETED IS -->) &ITG
MAXTG=TG
ITG=&ITG
TG=ITG+10
&TG=TG
<^U>
DELETE ENTITY: TAG[&TG]<CR>
XDEL=(FWD(ITG)+AFT(ITG))/2.
YDEL=5
DELETE ENTITY: X[XDEL]Y[YDEL]<CR>
REPAINT
SELECT MODE DRAW
SELECT LAYER 200
YLOC=8-.3125*ITG
DELETE ENTITY: X3.30Y[YLOC]<CR>
DELETE ENTITY: X5.30Y[YLOC]<CR>
DELETE ENTITY: X7.30Y[YLOC]<CR>
DELETE ENTITY: X9.30Y[YLOC]<CR>
DELETE ENTITY: X11.5Y[YLOC]<CR>
DELETE ENTITY: X13.3Y[YLOC]<CR>
<#----------------------------
<# UPDATE SUMMARIES
<#----------------------------
PVOL=PVOL-VOL(ITG)
RVOL=TVOL-PVOL
IF (RVOL.LT.0) NFLG=1
IF (NFLG.EQ.1) RVOL=ABS(RVOL)
&PVOL=PVOL
&RVOL=RVOL
MOMENT=MOMENT-VOL(ITG)*XCB(ITG)
LCB=MOMENT/PVOL
LCB=(AINT((LCB+.05)*10))/10
&LCB=LCB
DELETE ENTITY: X20.25Y7.6875<CR>
INSERT TEXT [&PVOL] HGT.1875 RJT: X20.25Y7.6875<CR>
DELETE ENTITY: X20.25Y7.375<CR>
INSERT TEXT [&RVOL] HGT.1875 RJT: X20.25Y7.375<CR>
IF (NFLG.EQ.1) ECHO LAYER INCLUDE 205
IF (NFLG.EQ.0) ECHO LAYER EXCLUDE 205
DELETE ENTITY: X20.25Y6.75<CR>
INSERT TEXT [&LCB] HGT.1875 RJT: X20.25Y6.75<CR>
IF (NFLG.EQ.1) RVOL=-RVOL
IF (NFLG.EQ.1) NFLG=0
PDF(ITG)=0
PDA(ITG)=0
FWD(ITG)=0
AFT(ITG)=0
VOL(ITG)=0
```

```
XCB(ITG)=0
LEN(ITG)=0
<#------------------------------
<# REPLACE DELETED SEGMENT
<#------------------------------
SELECT MODE MODEL
SELECT LAYER 2
<^U>
EXECV
#MENU2
PRNT
PRNT SELECT FORM OF PRESSURE HULL SEGMENT
PRNT 1-CYLINDER          2-CONICAL SEGMENT    3-FLAT PLATE CLOSURE
PRNT 4-SPHERICAL CLOSURE 5-ELLIPTICAL CLOSURE 6-SONAR AND ACCESS TRUNK
READ (7-OBTAIN DIAMETER   R-RETURN MAIN MENU    X-LEAVE -->) &TYPE
IF (&TYPE.EQ."1") GOSUB CYLIN
IF (&TYPE.EQ."2") GOSUB CONE
IF (&TYPE.EQ."3") GOSUB PLATE
IF (&TYPE.EQ."4") GOSUB SPHI
IF (&TYPE.EQ."5") GOSUB ELLIP
IF (&TYPE.EQ."6") GOSUB SONAR
IF (&TYPE.EQ."7") GOSUB GETR
IF (&TYPE.EQ."X") GOSUB STP
TPE(ITG)=&TYPE
TG=MAXTG-1
RTNSUB
<#********************************
<# PLOT HARD COPY
<#********************************
#PLT
PRNT
PRNT
PRNT PLOT ROUTINE
READ (DO YOU WISH TO PLOT DRAWING AS IS?  -->) &TYPE
IF (&TYPE.EQ."Y") PLOT DOT SCALE .5
EXECV
PRNT
READ (DO YOU WISH TO PLOT ISOMETRIC VIEW?  -->) &TYPE
IF (&TYPE.EQ."Y") GOSUB PLTISO
RTNSUB
<#------------------------------
<# PLTISO
<#------------------------------
#PLTISO
ECHO LAYER EXCLUDE 200
UNBLANK VIEW NAME 'ISO'
PLOT DOT SCALE .5
BLANK VIEW NAME 'ISO'
ECHO LAYER INCLUDE 200
RTNSUB
<#**************************************
<# DETERMINE DIAMETER AT GIVEN LOCATION
<#**************************************
#GETR
EXECV
PRNT
PRNT
PRNT
READ (LOCATION AT WHICH DIAMETER IS REQUIRED --> ) XL
GOSUB DRAD
```

```
PRNT
PRNT
PRNT
PRNT ENVELOPE DIAMETER AT [XL] IS [2*RAD1] FEET
I=1
REPEAT
  I=I+1
UNTIL (I.EQ.600)
RTNSUB
<#------------------------------
<#CALCULATE RADIUS
<#------------------------------
#DRAD
IF (XL.LE.LF) GOTO FRONT
IF (XL.GT.(LOA-LA)) GOTO AFT
IF (XL.GT.LF.AND.XL.LE.(LOA-LA)) RAD1=RAD
RTNSUB
#FRONT
XX=LF-XL
RAD1=RAD*(1-(XX/LF)**NF)**(1/NF)
RTNSUB
#AFT
XX=XL-(LOA-LA)
RAD1=RAD*(1-(XX/LA)**NA)
RTNSUB
<#********************************
<# CHECK FOR EXCESSIVE RADIUS
<#********************************
#CHECK
XL=FWD(ITG)
GOSUB DRAD
IF (RAD1.GE.(PDF(ITG)/2)) ERR1=0
IF (RAD1.LT.(PDF(ITG)/2)) ERR1=1
XL=AFT(ITG)
GOSUB DRAD
IF (RAD1.GE.(PDA(ITG)/2)) ERR2=0
IF (RAD1.LT.(PDA(ITG)/2)) ERR2=1
IF (ERR1.EQ.0.AND.ERR2.EQ.0) RTNSUB
PRNT
PRNT
PRNT
PRNT
IF (ERR1.EQ.1) PRNT ***ERROR*** FORWARD DIAMETER EXCEEDS ENVELOPE
IF (ERR2.EQ.1) PRNT ***ERROR*** AFT DIAMETER EXCEEDS ENVELOPE
RTNSUB
<#********************************
<# EXIT MIT.PHULL
<#********************************
#END
PRNT
PRNT
READ (DO YOU WISH TO MAKE A COPY OF THE DRAWING? ) &TYPE
IF (&TYPE.EQ."Y") GOSUB PLT
<#------------------------------
<# SAVE DATA
<#------------------------------
#SAVE
EXECV
PRNT
PRNT
```

```
PRNT
PRNT SAVING PRESSURE HULL DATA...
OPENW 1,"MIT.PASSFWD"
OPENW 2,"MIT.PASSAFT"
OPENW 3,"MIT.PASSDFWD"
OPENW 4,"MIT.PASSDAFT"
OPENW 5,"MIT.PASSVOL"
OPENW 6,"MIT.PASSXCB"
OPENW 7,"MIT.PHDATA"
OPENW 8,"MIT.PASSTYPE"
I=1
REPEAT
  &TYPE=FWD(I)
  WRITEF 1,&TYPE
  &TYPE=AFT(I)
  WRITEF 2,&TYPE
  &TYPE=PDF(I)
  WRITEF 3,&TYPE
  &TYPE=PDA(I)
  WRITEF 4,&TYPE
  &TYPE=VOL(I)
  WRITEF 5,&TYPE
  &TYPE=XCB(I)
  WRITEF 6,&TYPE
  &TYPE=TPE(ITG)
  WRITEF 8,&TYPE
  I=I+1
UNTIL (I.GT.20)
&TYPE=TG
WRITEF 7,&TYPE
&TYPE=TVOL
WRITEF 7,&TYPE
&TYPE=RES
WRITEF 7,&TYPE
&TYPE=PVOL
WRITEF 7,&TYPE
&TYPE=RVOL
WRITEF 7,&TYPE
&TYPE=LCB
WRITEF 7,&TYPE
<#-----------------------------
<# DETERMINE NEXT ACTION
<#-----------------------------
EXECV
PRNT
PRNT
PRNT
READ (DO YOU WISH TO ADD APPENDAGES AT THIS TIME? ) &TYPE
IF (&TYPE.EQ."Y") RUN NEW MIT.SAILI
END
```

```
MIT.&BCD.SAIL1
 5- 7-84 11:39:54 FUTIL 6.21

<#**********************************
<# MIT.SAIL1
<#**********************************
<#
<#  GENERATES DRAWING OF SAIL
<#  PREPARES FOR ENTRY TO BALANCE
<#
<#**********************************
DIM SX(28),SY(28),SZ(28)
DIM AFT(20),FWD(20),PDA(20),PDF(20),LEN(20),VOL(20),XCB(20)
DIM TPE(20)
<#**********************************
<# OBTAIN DATA FROM MIT.PASSPHI
<#**********************************
EXECV
PRNT
PRNT
PRNT
PRNT OBTAINING DATA FROM PASS FILES...
OPENR 1,"MIT.PASSPHI"
READF 1,&TYPE
DIAM=&TYPE
READF 1,&TYPE
LOA=&TYPE
READF 1,&TYPE
LF=&TYPE
READF 1,&TYPE
LA=&TYPE
READF 1,&TYPE
READF 1,&TYPE
READF 1,&TYPE
READF 1,&TYPE
NF=&TYPE
READF 1,&TYPE
NA=&TYPE
RAD=DIAM/2
<#------------------------------
<# LOCATE SAIL
<#------------------------------
PRNT
PRNT
PRNT
READ (FORWARD EDGE OF SAIL IS LOCATED AT--> ) SFD
GOSUB SAIL
GOTO END
<#**********************************
<# DRAW SAIL
<#**********************************
#SAIL
SELECT MODE MODEL
PRNT
PRNT
PRNT
PRNT OBTAINING SAIL OFFSETS...
<^U>
OPENR 1,"MIT.SAIL"
```

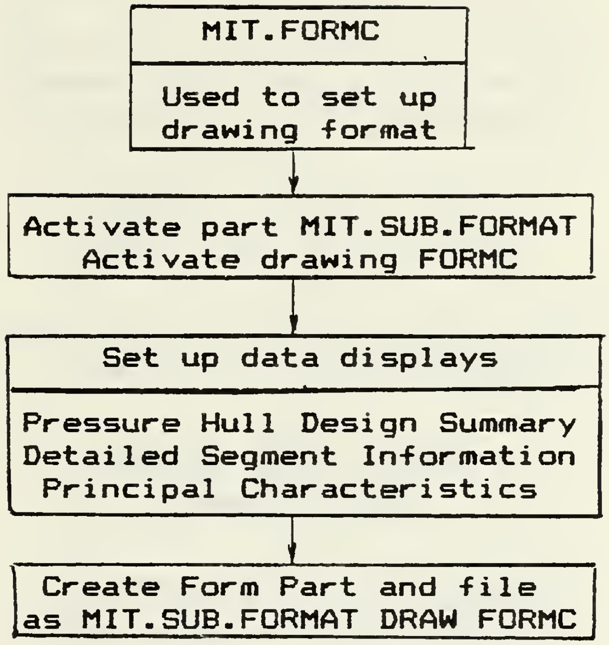
```
<#-----------------------------
<# READ OFFSETS
<#-----------------------------
READF 1,&TYPE
SLEN=&TYPE
READF 1,&TYPE
HT=&TYPE
I=1
REPEAT
  READF 1,&TXT
  SXX=&TXT
  SX(I)=SXX+SFD
  READF 1,&TXT
  SZ(I)=&TXT
  I=I+1
UNTIL (I.EQ.29)
<#-----------------------------
<# DETERMINE INTERSECTION POINTS
<#-----------------------------
I=1
REPEAT
  XL=SX(I)
  GOSUB DRAD
  SR=RAD1
  SY(I)=SQRT(SR**2-SZ(I)**2)
  I=I+1
UNTIL (I.EQ.29)
<#-----------------------------
<# DRAW CURVE OF INTERSECTION
<#-----------------------------
I=1
TTG=90
&TG=TTG
SELECT LAYER 5
INSERT BSPLINE INTR TAG=[&TG]:<#
#INCA X[SX(I)]Y[SY(I)]Z[SZ(I)],<#
I=I+1
IF (I.LE.28) GOTO INCA
<CR>
I=1
TTG=92
&TG=TTG
INSERT BSPLINE INTR TAG=[&TG]:<#
#INCC X[SX(I)]Y[SY(I)]Z[-SZ(I)],<#
I=I+1
IF (I.LE.28) GOTO INCC
<CR>
<#-----------------------------
<# DRAW UPPER BOUNDARY
<#-----------------------------
I=1
TTG=91
&TG=TTG
YT=RAD+HT
INSERT BSPLINE INTR TAG=[&TG]:<#
#INCB X[SX(I)]Y[YT]Z[SZ(I)],<#
I=I+1
IF (I.LE.28) GOTO INCB
<CR>
I=1
```

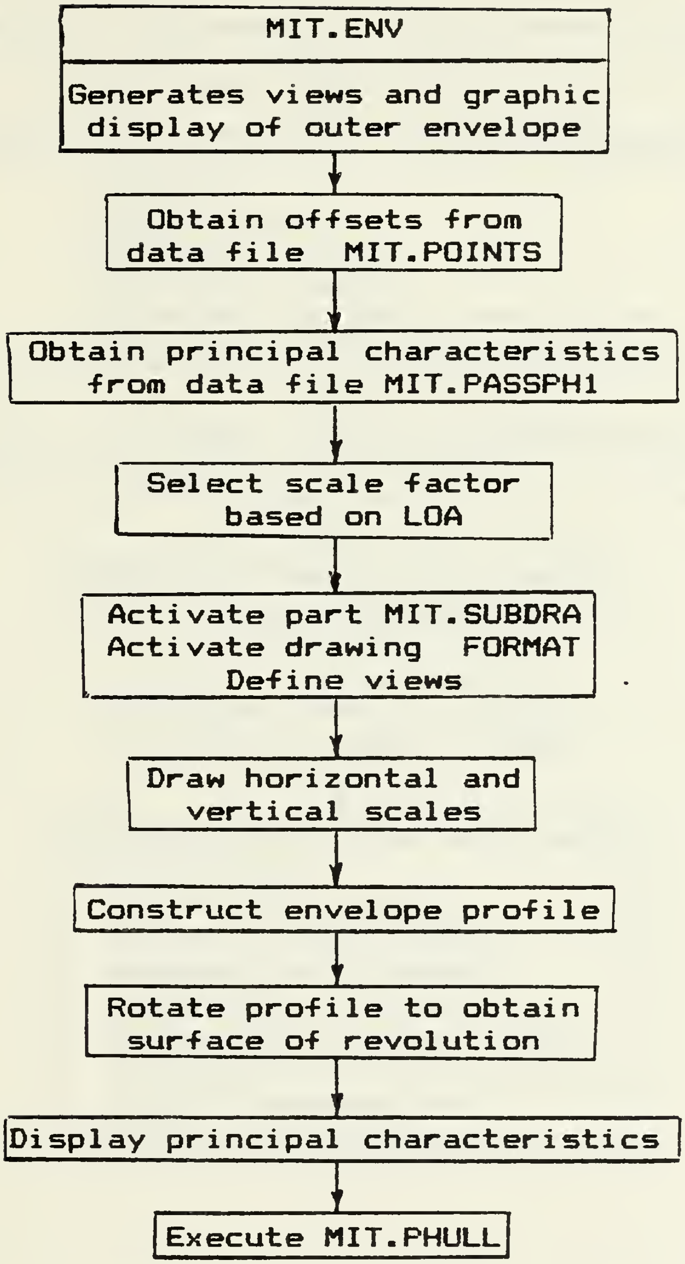
```
TTG=93
&TG=TTG
INSERT BSPLINE INTR TAG=[&TG]:<#
#INCD X[SX(I)]Y[YT]Z[-SZ(I)],<#
I=I+1
IF (I.LE.28) GOTO INCD
<CR>
<#------------------------------
<# INSERT SURFACE
<#------------------------------
&TG=90
&TG1=91
INSERT RSURFACE MESH6X6: TAG[&TG],TAG[&TG1]<CR>
&TG=92
&TG1=93
INSERT RSURFACE MESH6X6: TAG[&TG],TAG[&TG1]<CR>
<^U>
RTNSUB
<#*************************************
<# DETERMINE RADIUS AT GIVEN LOCATION
<#*************************************
#DRAD
IF (XL.LE.LF) GOTO FRONT
IF (XL.GT.(LOA-LA)) GOTO AFT
IF (XL.GT.LF.AND.XL.LE.(LOA-LA)) RADI=RAD
RTNSUB
#FRONT
XX=LF-XL
RADI=RAD*(1-(XX/LF)**NF)**(1/NF)
RTNSUB
#AFT
XX=XL-(LOA-LA)
RADI=RAD*(1-(XX/LA)**NA)
RTNSUB
<#*********************************
<# PLOT HARD COPY
<#*********************************
#PLT
PRNT
PRNT
PRNT PLOT ROUTINE
READ (DO YOU WISH TO PLOT DRAWING AS IS?  -->) &ANS
IF (&ANS.EQ."Y") PLOT DOT SCALE .5
EXECV
PRNT
READ (DO YOU WISH TO PLOT ISOMETRIC VIEW?  -->) &ANS
IF (&ANS.EQ."Y") GOSUB PLTISO
RTNSUB
<#------------------------------
<# PLTISO
<#------------------------------
#PLTISO
ECHO LAYER EXCLUDE 200
UNBLANK VIEW NAME 'ISO'
PLOT DOT SCALE .5
RTNSUB
<#*********************************
<# CHANGE MESH
<#*********************************
#CLEAR
```

```
<#------------------------------
<# RETRIEVE PRESSURE HULL DATA
<#------------------------------
PRNT
PRNT
PRNT
PRNT RETRIEVING PRESSURE HULL DATA
<^U>
OPENR 1,"MIT.PASSFWD"
OPENR 2,"MIT.PASSAFT"
OPENR 7,"MIT.PHDATA"
I=1
REPEAT
  READF 1,&TYPE
  FWD(I)=&TYPE
  READF 2,&TYPE
  AFT(I)=&TYPE
  I=I+1
UNTIL (I.GT.20)
READF 7,&TYPE
TGM=&TYPE
TG=11
REPEAT
  ITG=TG-10
  X1=(FWD(ITG)+AFT(ITG))/2
  Y1=3
  CHANGE SGRAPHICS MESH1X2 : X[X1]Y[Y1]<CR>
  TG=TG+1
UNTIL (TG.GT.TGM)
<#------------------------------
<# SAIL MESH
<#------------------------------
CHANGE SGRAPHICS MESH 1X2 : X[SFD+SLEN/2]Y[RAD+HT/2]
CHANGE SGRAPHICS MESH 1X2 : X[SFD+SLEN/2]Y[RAD+HT/2]
<^U>
RTNSUB
<#********************************
<# EXIT MIT.SAIL1
<#********************************
#END
EXECV
PRNT
PRNT
READ (DO YOU WISH TO MAKE A COPY OF THE DRAWING?  ) &ANS
IF (&ANS.EQ."Y") GOSUB PLT
EXECV
PRNT
PRNT
PRNT
READ (DO YOU WISH TO CONTINUE TO BALANCE?  ) &ANS
IF (&ANS.EQ."Y") GOSUB CLEAR
IF (&ANS.EQ."Y") RUN NEW MIT.BALANCE
END
```

APPENDIX III

PROGRAM FLOWCHARTS


MIT.FORMC
Used to set up drawing format
Activate part MIT.SUB.FORMAT
Activate drawing FORMC
Set up data displays
Pressure Hull Design Summary
Detailed Segment Information
Principal Characteristics
Create Form Part and file as MIT.SUB.FORMAT DRAW FORMC


MIT.ENV
Generates views and graphic display of outer envelope
Obtain offsets from data file MIT.POINTS
Obtain principal characteristics from data file MIT.PASSPH1
Select scale factor based on LOA
Activate part MIT.SUBDRA
Activate drawing FORMAT
Define views
Draw horizontal and vertical scales
Construct envelope profile
Rotate profile to obtain surface of revolution
Display principal characteristics
Execute MIT.PHULL

MIT.PHULL

Create 3D representation of pressure hull, calculate and display various parameters

↓

Obtain the following from MIT.PASSPH1:
- Diameter
- Length overall
- Forebody length
- Afterbody length
- Forebody fullness coefficient
- Afterbody fullness coefficient

↓

Obtain the following from MIT.PASSPH2:
- A weight

↓

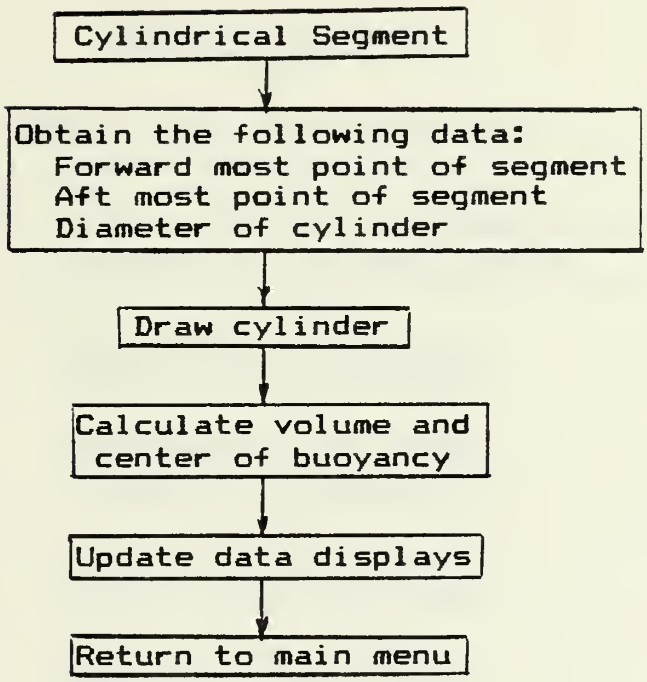
Determine amount of A weight to be reserved

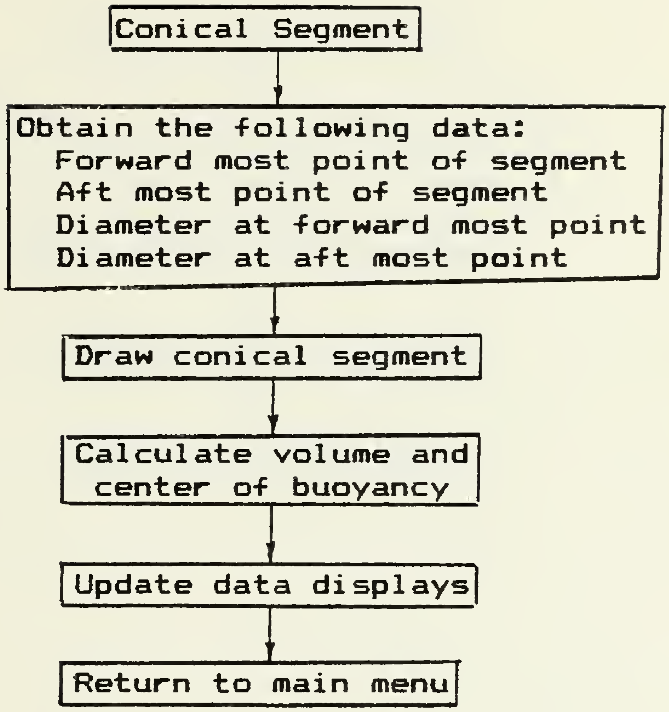
↓

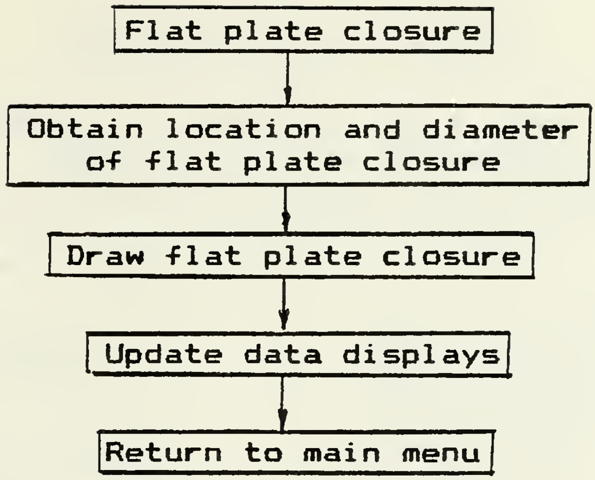
Display initial values in Design Summary area

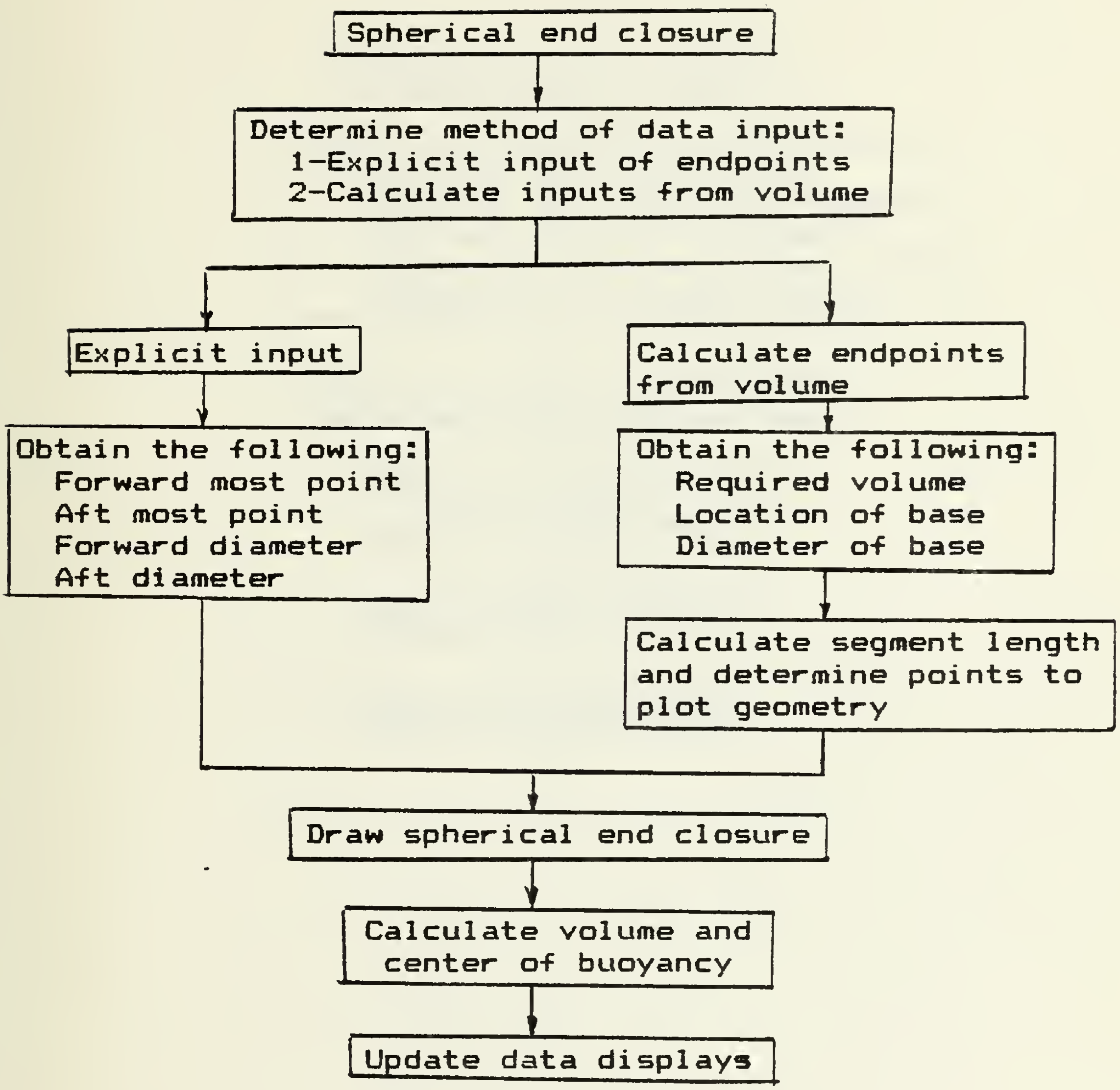
↓

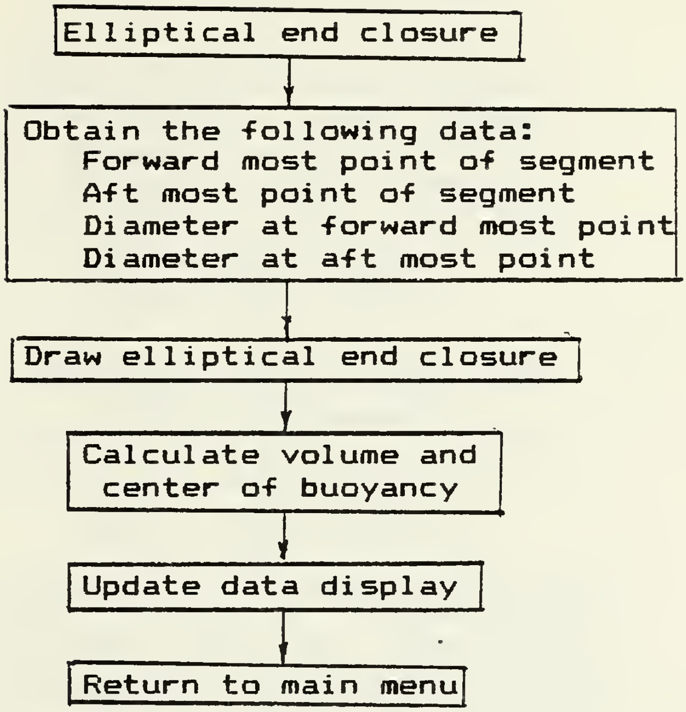
Select option from main menu
- 1-Cylinder
- 2-Conical segment
- 3-Flat plate closure
- 4-Spherical closure
- 5-Elliptical closure
- 6-Sonar and access trunk
- 7-Diameter at any location
- E-Edit pressure hull
- X-Leave MIT.PHULL

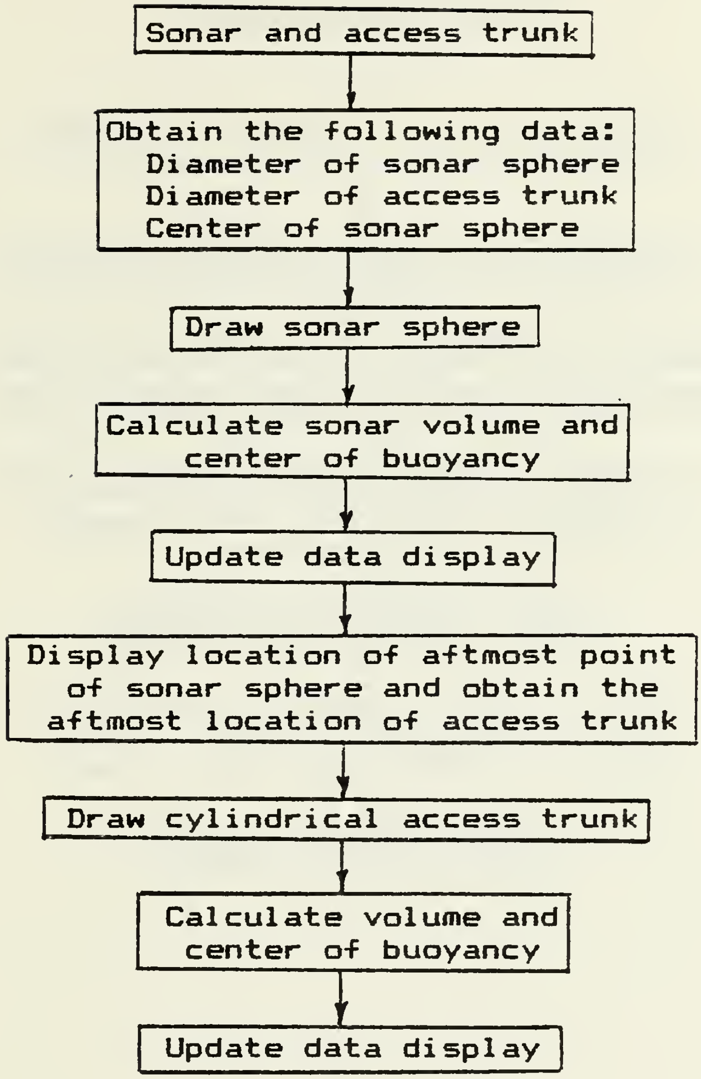

Cylindrical Segment
Obtain the following data:
Forward most point of segment
Aft most point of segment
Diameter of cylinder
Draw cylinder
Calculate volume and center of buoyancy
Update data displays
Return to main menu

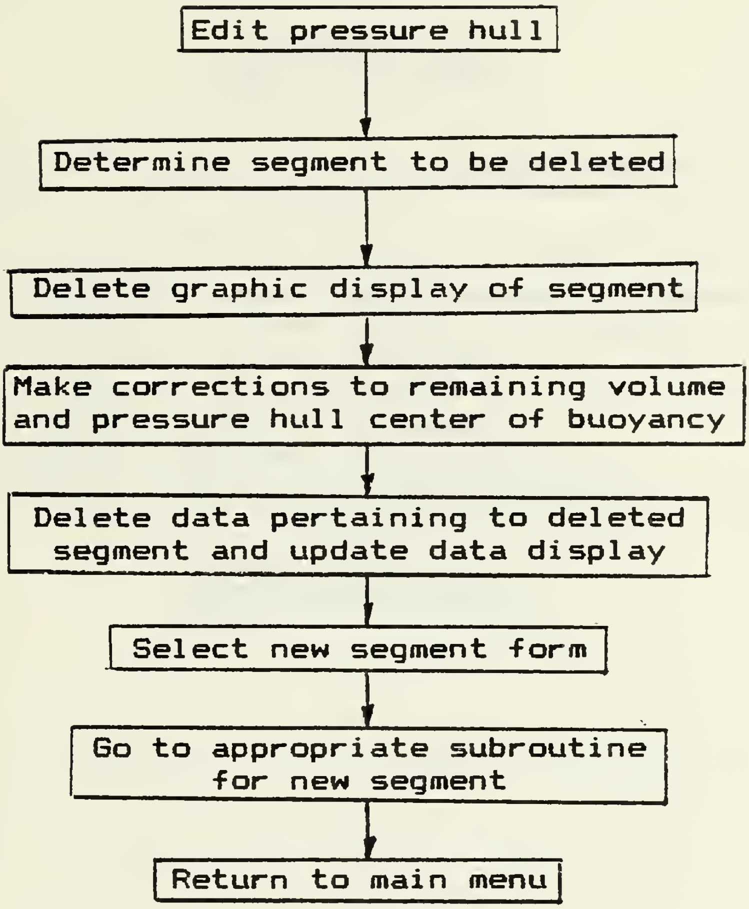

Conical Segment
Obtain the following data:
Forward most point of segment
Aft most point of segment
Diameter at forward most point
Diameter at aft most point
Draw conical segment
Calculate volume and center of buoyancy
Update data displays
Return to main menu


Flat plate closure
Obtain location and diameter of flat plate closure
Draw flat plate closure
Update data displays
Return to main menu


Spherical end closure
Determine method of data input:
1-Explicit input of endpoints
2-Calculate inputs from volume
Explicit input
Calculate endpoints from volume
Obtain the following:
Forward most point
Aft most point
Forward diameter
Aft diameter
Obtain the following:
Required volume
Location of base
Diameter of base
Calculate segment length and determine points to plot geometry
Draw spherical end closure
Calculate volume and center of buoyancy
Update data displays


Elliptical end closure
Obtain the following data:
Forward most point of segment
Aft most point of segment
Diameter at forward most point
Diameter at aft most point
Draw elliptical end closure
Calculate volume and center of buoyancy
Update data display
Return to main menu


Sonar and access trunk
Obtain the following data:
Diameter of sonar sphere
Diameter of access trunk
Center of sonar sphere
Draw sonar sphere
Calculate sonar volume and center of buoyancy
Update data display
Display location of aftmost point of sonar sphere and obtain the aftmost location of access trunk
Draw cylindrical access trunk
Calculate volume and center of buoyancy
Update data display


Edit pressure hull
Determine segment to be deleted
Delete graphic display of segment
Make corrections to remaining volume and pressure hull center of buoyancy
Delete data pertaining to deleted segment and update data display
Select new segment form
Go to appropriate subroutine for new segment
Return to main menu

Exit MIT.PHULL

↓

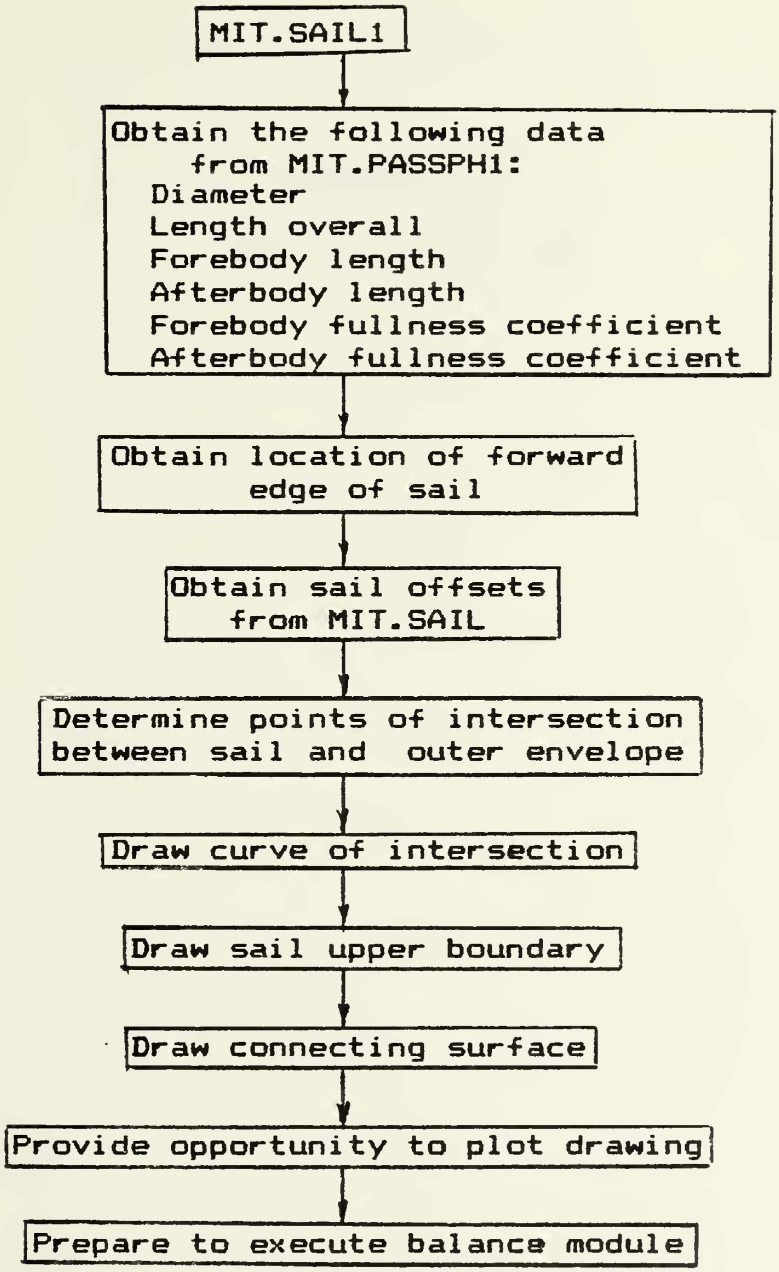
Provide opportunity to plot drawing

↓

Save data in the following data files:
MIT.PASSFWD: forward most points of segments
MIT.PASSAFT: aftmost points of segments
MIT.PASSDFWD: forward diameters
MIT.PASSDAFT: aft diameters
MIT.PASSVOL: segment volumes
MIT.PASSXCB: centers of buoyancy
MIT.PHDATA: totals for pressure hull

↓

Execute MIT.SAIL1


MIT.SAIL1
Obtain the following data from MIT.PASSPH1:
Diameter
Length overall
Forebody length
Afterbody length
Forebody fullness coefficient
Afterbody fullness coefficient
Obtain location of forward edge of sail
Obtain sail offsets from MIT.SAIL
Determine points of intersection between sail and outer envelope
Draw curve of intersection
Draw sail upper boundary
Draw connecting surface
Provide opportunity to plot drawing
Prepare to execute balance module

APPENDIX IV

PROGRAM DATA FILES

DATA FILES USED BY GRAPHICS MODULES

1. Files created by earlier modules.

MIT.POINTS: This data file is used in MIT.ENV to obtain the 132 offsets generated in the geometry module.

MIT.PASSH1: Used by MIT.ENV, MIT.PHULL, and MIT.SAIL1. This data file contains the following data concerning the envelope geometry:

- Diameter
- Length overall
- Length of forebody
- Length of afterbody
- Submerged center of buoyancy
- Submerged displacement
- Envelope displacement
- Forebody fullness coefficient
- Afterbody coefficient

MIT.PASSPH2: This data file is used by MIT.PHULL to obtain the A weight and design name.

2. Files created by graphics modules.

a. Data files are created in MIT.PHULL to save data describing the individual segments of the pressure hull as well as the total volume and center of buoyancy of the pressure hull. Within this module, all data concerning each segment is maintained in a series of one dimensional arrays. Prior to leaving the module this data is saved in a series of data files.

MIT.PASSFWD: Contains locations of the forward most point of each segment.

MIT.PASSAFT: Contains locations of the aftmost point of each segment.

MIT.PASSDFWD: Contains diameter at the forward most point of each segment.

MIT.PASSDAFT: Contains the diameter at the aftmost point of each segment.

MIT.PASSVOL: Volume of each pressure hull segment.

MIT.PASSLCB: Location of center of buoyancy of each segment.

MIT.PASSTYPE: Contains the type of each segment (cone, cylinder, etc.)

MIT.PHDATA: Contains total volume and center of buoyancy of the pressure hull.

b. Parts and drawings generated by graphics modules.

MIT.SUB.FORMAT: This is the part used by MIT.FORMC to generate the Form Part drawing. MIT.SUB.FORMAT.FORMC is the name of the drawing.

MIT.SUBDRA: This part is activated in MIT.ENV. It is used in all graphics modules.

MIT.HULLOUT: Upon completion of CADSUB, the part MIT.SUBDRA is filed under this name.

APPENDIX V

LAYER ASSIGNMENTS

LAYER ASSIGNMENTS

Through the use of layers one may remove from view parts of a design without totally eliminating them. This makes it possible to improve the clarity of the drawing and view only those elements of the design that are required at any time in the design process.

The following layer assignments are used in CADSUB graphics modules.

Layer 0: Border of drawing, horizontal and vertical scales.

Layer 1: Outer envelope.

Layer 2: Pressure hull.

Layer 5: Sail.

Layer 199: Principal characteristics data display.

Layer 200: Data display areas for detailed segment information and pressure hull design summary.

Layer 205: Negative sign. This is included because of the undesirable effects observed in MIT.PHULL when attempting to convert a negative remaining volume to a text entity and including it in the data display. If this value is negative, a flag is set and the value is converted to a positive value. If, when written on the screen, the flag is set Layer 205 is included in the drawing. If the flag is not set Layer 205 is excluded from view.

Thesis 207077
M39954 Meade
c.1 Application of computer aided design to submarines.

Thesis 207077
M39954 Meade
c.1 Application of computer aided design to submarines.

thesM39954
Application of computer aided design to
3 2768 001 88557 7
DUDLEY KNOX LIBRARY